An Integrity Framework for Image-Based Navigation Systems

DISSERTATION

Craig D. Larson, Captain, USAF

AFIT/DEE/ENG/10-03

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

The views expressed in this dissertation are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States

AFIT/DEE/ENG/10-03

AN INTEGRITY FRAMEWORK FOR IMAGE-BASED NAVIGATION SYSTEMS

DISSERTATION

Presented to the Faculty
Graduate School of Engineering and Management
Air Force Institute of Technology
Air University
Air Education and Training Command
in Partial Fulfillment of the Requirements for the
Degree of Doctor of Philosophy

Craig D. Larson, B.S.E.E., M.S.E.E.
Captain, USAF

June 2010

AFIT/DEE/ENG/10-03

AN INTEGRITY FRAMEWORK FOR IMAGE-BASED NAVIGATION SYSTEMS

Craig D. Larson, B.S.E.E., M.S.E.E.
Captain, USAF

Approved:

	3 MAY 2010
John F. Raquet, PhD Chairman	Date
	21 APRIL 2010
Lt Col Michael J. Veth, PhD Member	Date
Mark E. Oxley	20 April 2010
Mark E. Oxley, PhD Member	Date
Frank van Graas	20 April 2010
Frank van Graas, PhD Member	Date

Accepted:

MU Thomas	1 Jun 2010
M. U. Thomas Dean, Graduate School of Engineering and Management	Date

AFIT/DEE/ENG/10-03

Abstract

The value of Global Navigation Satellite Systems (GNSS) in a multitude of both military and civilian navigation and timing applications has been indisputably shown over the last twenty plus years. Increased dependence on GNSS has driven the need for risk management in using these systems for critical operations. One example risk reduction is through use of an integrity monitoring process, a process which predicts the impact of a worst case or minimal detectable bias error on the navigation position error. The navigation community has spent nearly two decades in development of GNSS integrity algorithms and continues to refine methodologies and investigate new approaches. As a result, the concept of integrity has been formalized for GPS to encompass fundamental criteria, including the ability to detect measurement errors and notify the user when the system should not be used because of potentially corrupt measurements.

More recently, researchers have begun investigating the potential use of image-based navigation systems for applications where GNSS signals are not available or should not be used. Independence from external radio-frequency transmissions and potentially low-cost implementations of camera and image processing suites are among the key advantages of these newer systems. However, while larger issues of integrity are seemingly well-understood for many GNSS applications, to the best of our knowledge, no one has fully addressed rigorous integrity quantification with respect to image-based navigation systems.

This dissertation first examines and addresses the fundamental differences existing between the measurement models established for GPS and those of proposed image-based navigation systems. For example, GPS pseudorange measurements used in a position determination are provided as a single value per satellite. However, image measurements are inherently angle-based measurements and assumed to be provided as a pair of pixel coordinates for a mapped target. Under these conditions, consideration must be given not only to the units of the transformations between the states and measurements, but also to the fact that multiple rows of the observation matrix relate to particular error states in the image-based case. This is due the added dimensionality of the pixel pair measurements.

Based on this examination, an algorithm is then developed to instantiate a framework for image-based integrity analogous to that of GPS RAIM, as to be understood in common terms by the navigation community. The algorithm is applied to the case where the navigation system is estimating position only and then extended to the case where estimation of both position and attitude is required. Detailed analysis is presented demonstrating the impact of angular error on a single pixel pair measurement, comparing results from both estimation scenarios, and showing that, from an integrity perspective, there is a significant benefit of having known attitude information. Additional studies are done to demonstrate the impact of pixel pair measurement relative geometries on system integrity in this framework, showing potential improvement in image-based integrity through screening and adding measurements to the navigation system solution, when additional measurements are available.

Acknowledgements

I am forever grateful for the immeasurable support and encouragement I have received throughout this endeavor. I cannot possibly, in the allowable space, thank all of the people who have touched my life in a such a positive manner, and in ways they may not have even realized. I can say that I am blessed and am better for having known them.

I must personally express my deepest appreciation for those most responsible in helping me accomplish this goal. First and foremost, I am thankful for my faith and family, and I could not have made it through without the unwavering support of both. I am sincerely grateful to my advisor, Dr. Raquet, for his unmatched insights, invaluable mentoring, and incredible patience. I would also like to thank my committee members, Lt Col Veth, Dr. van Graas, and Dr. Oxley. In addition to being outstanding as individuals and as a team, their wealth of knowledge, talent, and varied experience was critical to this journey and helped me grow on both a personal and professional level. Thank you also to my fellow classmates and AFIT staff whose timely words of wisdom and esprit de corps made the trip worth taking. I would also like to thank the U.S. Air Force for providing me the time and opportunity to pursue my education.

Craig D. Larson

Table of Contents

Page

List of Figures

Figure Page

Figure Page

Figure Page

List of Tables

Table Page

An Integrity Framework for Image-Based Navigation Systems

I. Introduction

Modern military aerospace operations demand highly accurate navigation systems to faithfully execute their missions. Examples of such missions include precision bombing, aerial refueling, or simply maintaining a designated spacing in a route corridor, with the latter being an important consideration for civilian air operations as well. The advent of Global Positioning System (GPS) positioning has brought unprecedented levels of navigation accuracy to both the military and civilian communities. With the addition of new Global Navigation Satellite Systems (GNSS) being developed or improved, the ability to accurately determine position anywhere on the globe is likely to get even better. The degree of accuracy that can be achieved is based on how well the navigation system estimates the true position under normal operating conditions and the quality of sensor measurements provided to the system. Regardless of the navigation system employed, a wide variety of equipment, environmental, or modeling errors affecting the system prevent a perfect determination of the true position.

For example, an Inertial Navigation System (INS) uses specific force and angular rate measurements to generate estimates of vehicle position, velocity, and attitude as a result of vehicle dynamics. The actual measurements for the INS are produced by accelerometers and gyroscopes, respectively, jam-resistant sensors that are unaffected by radio-frequency interferers. This non-transmitting/non-receiving nature of an INS makes it very useful in military applications, where navigators desire to avoid enemy detection. Unfortunately, these sensors are influenced by physical errors, such as gravity deflections, as well as misalignments, drifts, and biases stemming from material or design limitations.

The impact of these errors grow over time through the process of integrating the measurements to produce the navigation estimates, degrading system accuracy. Because of this degradation, applications requiring a higher degree of long-term accuracy often integrate an INS with another navigation system such as GPS. Given a sufficient number of satellites in view, GPS measurements provide additional position (and possibly velocity) information to the integrated system, helping to bound the INS errors and improve navigation performance of the system. However, unlike INS, GPS measurements, and those of other GNSS, are provided via radio-frequency signals, which are susceptible to signal obstruction, as well as sources of intentional and unintentional interference.

This vulnerability in radio-frequency signals has driven recent research into alternative navigation schemes to reduce or eliminate reliance on GNSS, including the fusion of an INS with camera systems [28; 35; 42; 51; 59; 78; 86; 99; 106; 115; 119; 121; 122; 123; 126; 127]. Image-aided navigation holds potential not only for improved interference immunity, but also allows reduced detectability of the user because it is passive, like INS. An additional advantage for image-aided navigation systems is that cameras acceptable for this application are potentially inexpensive, can be very small, and have minimal power requirements [99]. The cited research has shown the feasibility of their respective integrated approaches and demonstrated reasonable levels of accuracy, although not completely quantified, on the order of meters in some cases. The achievable level of accuracy in an integrated INS/image-aided system is a function of the previously mentioned INS error sources and imaging system considerations including the camera equipment, feature tracking algorithm, the vehicle trajectory, and the image scene [28].

With image-based navigation systems quickly developing into a capable augmentation, or even replacement in some instances, of GPS positioning in the aiding of an INS, means of assuring navigation accuracy become necessary. This dissertation details the research effort to develop a rigorous integrity approach for image-aided navigation. The research is motivated by the absence of clearly defined integrity monitoring algorithms and qualitative integrity metrics for assuring image-aided navigation.

1.1 Components of Navigation System Performance

Accuracy alone, however, is an insufficient criterion to define system performance in precise navigation. Since the true position can never be absolutely known, accuracy is based on the assumption that the navigation system is operating within specifications, i.e. no faults are present. But what if this no-fault assumption is false? Electrical or mechanical equipment failures may be easily identifiable through built-in monitoring systems, but a more subtle and potentially hazardous condition exists when a “soft” failure permits continued operation of a navigation system whose navigation solution has been corrupted by the incorporation of faulty sensor data. An example of this is when a bias or ramp error occurs in a GPS pseudorange measurement and this measurement is used to compute position. The results could be catastrophic, if, for example, two aircraft failed to maintain a safe separation due to a misleading position solution. Therefore, the navigation system should provide not only an estimate of position, but also ensure that the position estimate does not exceed specified limits without being identified; this concept is called integrity monitoring [25].

There are, in fact, four essential, and inextricably linked, components of navigation system performance. These four criteria are availability, continuity of function, integrity, and accuracy, and all are defined in the Federal Radionavigation Plan (FRP) [1]:

- ***Accuracy***: The degree to which the estimated position matches the true position, typically evaluated as a statistical measure of system error. Accuracy must be predictable and repeatable.
- ***Integrity*:** The ability of a system to provide a timely warning to users when a system should not be used for navigation.
- ***Continuity of function*:** The ability of the total system to function without interruption during an entire period of operation. This function includes the alarm rate, and

it is essentially the capability of a system to provide both navigation accuracy and integrity *throughout* the period of operation [94].

- ***Availability*:** The percentage of time the services of a navigation system are useable.

From these definitions, it is clear that a more precise definition of integrity is warranted for research in the area. This is an issue that will be addressed in following sections. For now, suffice it to say that integrity involves methods which, at a minimum, ensure errors are detected and reported. It is worth noting that it is possible to have poor accuracy with good integrity and good accuracy with poor integrity, since there are other factors to consider. However, the two are generally consistent, such that if integrity is poor, the accuracy is probably suspect. If accuracy and integrity requirements cannot be met, then the system should not be deemed available. It should be clear that these four components are tied together, and that a change in any one of them influences the status of the others. The focus of this research is on integrity, but in various discussions there will be inferences to the impact on the three other elements, because there are definite engineering trade-offs between all four components.

1.2 ***Focus on Integrity***

The definition of integrity given by the FRP also states that "Integrity is the measure of the trust that can be placed in the correctness of the information supplied by a navigation system" [1]. This statement relates the concept of integrity to that of accuracy. There are two fundamental criteria for integrity: 1) The ability to detect measurement errors, and 2) to notify the user that the system should not be used for navigation. These criteria are necessary for precise navigation systems, but only sufficient for supplemental navigation. In a supplemental system, the navigation system is required to provide a position estimate of a specified accuracy and identify, through integrity monitoring, when that accuracy cannot be met, allowing the navigator to terminate use and switch to an alternative system [25]. However, if the navigation system is to be used as the sole navigation positioning source (requiring no back-up system), the system must also be able recover from possible measurement errors. This means that the integrity method must not only detect errors, but also exclude the error source and prevent corruption of the navigation solution [25].

Detection of measurement errors can be done by using statistics based on prior analysis of such errors and evaluating them against a threshold based on a required level of performance. This last statement implies that outer performance limits must be assessed. Although not explicitly stated in the FRP statement above, the practical definition of integrity includes the ability to identify the upper bound on the level of measurement error that can be detected. Notification of a failure condition is made when measurement error falls outside the determined threshold, and must be done in a timely manner to allow the user to take action. There is a potential problem when a "bad" measurement, a measurement representing a fault or failure, falls within the threshold, since no alarm will be given. This condition is known as a missed detection. The tendency would then be to tighten the threshold such that fewer errors are missed. However, this gives rise to the potential for a measurement to fall outside the threshold when a failure has not actually occurred. This is condition is known as a false alarm.

The specified accuracy threshold and time to alarm are conditioned on the level of risk associated with the navigation application. Limiting factors must be considered. This includes accounting for the fundamental trade off in the ability to detect an error and the false alarm rate, since tightening the threshold to ensure a higher level of accuracy increases the number of false alarms, thereby limiting the availability of the system for navigation.

The navigation community has spent at least 20 years in development of GNSS integrity algorithms and continues to refine methodologies and investigate new approaches. To develop an image-aided integrity approach, it is necessary to examine in detail the foundational approaches and the current state-of-the-art techniques. The next section introduces a basic overview of the regulatory guidance driving GPS integrity to provide perspective on navigation system performance requirements.

1.3 Regulatory Background

This research focuses on integrity because the rapid growth in navigation technology over the last several decades has led to unprecedented levels of navigation position accuracy and continues to drive the desire for even greater precision. For example, the Federal Aviation Administration (FAA) has recognized that the development of advanced systems for determining position, including inertial reference units (IRU) and GNSS, now means air travel is no longer strictly limited to routes defined solely by ground-based radionavigation aids. Instead, navigation can be done via waypoints, using optimal routes, in a process the FAA calls area navigation (RNAV) [38].

The FAA has refined the definition of RNAV to establish performance metrics for airspace containment, requiring that navigation systems operate within navigation performance accuracy 95% of the time, while additionally maintaining continuity and integrity, in order to ensure aircraft containment certainty of 99.9% [38]. This standard is referred to as required navigation performance (RNP) by the FAA, and is purely performance based, not tied to specific equipment or platforms. The current drive is toward achieving a certainty level of 99.999%, maximizing both safety and efficiency in air travel [38].

However, the concept of RNP can be extended well beyond commercial air travel applications to all manners of operations where certainty of navigation system performance is critical. Close air operations, autonomous vehicle control, precision engagements, and air-to-air refueling are just a few areas where the concept has real application. If vision systems are used in safety-of-life situations in the future, like GPS is now, then integrity for them must be developed.

1.4 Key Definitions

The intent of the research is not only to develop an approach for image-aided integrity, but to do so in terms consistent with the GNSS community. In order to start framing an approach for INS/image system integrity, it is important to understand a number of key terms established by [67]. These are restated below, with several of the core definitions cited nearly verbatim:

Horizontal Alert Limit (HAL): The maximum horizontal position error allowed for a given navigation mode (e.g., en-route, non-precision approach) without an alarm being raised. In other words, the HAL is a specification. If a navigation system integrity algorithm cannot provide required assurances that the HAL will not be exceeded, then that navigation system is not allowed to be the primary system used in the navigation mode.

Horizontal Protection Limit (HPL): The horizontal position error that a fault detection and exclusion (FDE) algorithm guarantees will not be exceeded without being *detected* by the fault detection function based on missed detection and false alarm specifications and assumptions. For GPS, this is a function of only the visible satellites, user geometry, and expected error characteristics. Since it is not based on current measurements, the HPL can be predicted.

Horizontal Uncertainty Limit (HUL): The estimate of horizontal position uncertainty, based on inconsistencies in the measurements, that bounds the true error by $\geq 99.9\%$. Since actual measurements are required, the HUL must be computed in real-time (or may possibly be post-processed if recreating an event).

Horizontal Exclusion Limit (HEL): The horizontal position error that a fault detection and exclusion (FDE) algorithm guarantees will not be exceeded without being *detected and excluded* by the fault detection function based on missed detection and false alarm specifications. Again, for GPS, this is a function of only the visible satellites, user geometry, and expected error characteristics. However, the ability to exclude measurements in the HEL determination requires the availability of more independent measurements than mere detection.

Positioning Failure: A positioning failure occurs when the position solution error exceeds the applicable HPL (or HAL if the navigation mode is known by the user equipment).

False Alarm: The indication of a positioning failure when one has not occurred.

Missed Detection: A positioning failure not detected by the FDE algorithm.

Time to Alarm: Maximum time allowed between the onset of a positioning failure and the equipment alarm being initiated.

Incorrect Exclusion: An incorrect exclusion occurs when a failure is detected by the FDE algorithm, but the incorrect failure source is excluded. In the incorrect exclusion, failing to exclude the measurement may or may not cause a positioning failure.

Wrong Exclusion: A wrong exclusion is an incorrect exclusion resulting in a positioning failure and also potentially exceeding the required time-to-alarm for the navigation mode.

Failed Exclusion: A failed exclusion occurs when a valid failure is detected, but the detection condition is not resolved within the time-to-alarm from the onset of a positioning failure. A failed exclusion does not include the condition where the exclusion of the wrong failure source happens to resolve the detection condition.

This research focuses on horizontal positioning integrity in image-aided navigation and will use many of the definitions above. Figure 1.1 provides a visual example of the alert zones. It should be noted that, although not presented here, the terms for vertical positioning parallel those given above. With the basic definitions in place, the research problem can be formalized.

*1.5 **Problem Definition***

Limitations and vulnerabilities in GNSS have driven the navigation community to look for new alternatives in navigation, one of which is image-aided navigation. The

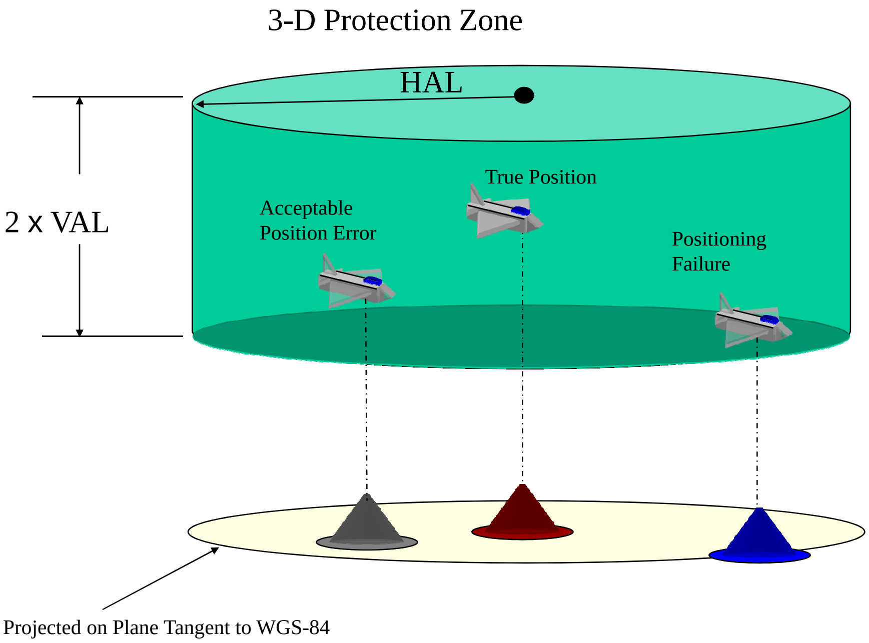
3-D Protection Zone
HAL
2 x VAL
True Position
Acceptable Position Error
Positioning Failure
Projected on Plane Tangent to WGS-84

Figure 1.1: Visualization of the alert zones defined by operating airspace; modeled on [89].

capability and viability of image-aided systems appears to be growing rapidly. As a result, it will be necessary to ensure that the position solutions provided by these systems provide the same level of confidence in safety critical applications that has come to be expected with GNSS-based systems. Therefore, the image-aided systems must meet the required components of navigation system performance, especially in terms of accuracy and integrity. However, no current research into the rigorous definition and implementation of integrity in image-aiding has been uncovered. The concept of navigation integrity has been well-studied in the GNSS community and continues to be a heavily researched area [8; 10; 11; 12; 15; 16; 18; 20; 23; 25; 29; 30; 31; 39; 67; 68; 69; 70; 71; 72; 95; 110; 129; 130], even after more than two decades of GPS usage. These concepts provide some foundation for the pursuit of image-aided integrity, but there are several fundamental differences between the two types of navigation systems that must be addressed to extend the concept of integrity to image-aiding.

GPS integrity is based on an inference of the accuracy of the position solution on the quality of the GPS measurements [25]. While this would also be the case for image-aided navigation, user estimates of the error sources associated with GPS are well documented and readily accessible. This is not the case for the much newer image-aided technology. Also, individual GPS measurements commonly used are range measurements, meaning that each measurement is provided by a single source, the satellite, and can be discriminated from other measurements at each time epoch by the transmitted satellite code. However, in the case of image-aiding, each measurement source is a target translated to an image plane, and is presented as a measurement in the form of a vector containing a pair of pixel locations. These pixel locations can be translated as two bearing measurements vice the single range measurement provided by GPS, for each source at each epoch. Common integrity methods for GPS have focused on single fault conditions due to the complexity of identifying and excluding multiple simultaneous faults, although current research is focusing on the multiple failure case. The potential for multiple failures in the image-aiding case is much higher than the GPS case due to the nature of the two-dimensional measurement.

Certainly, there have been *ad hoc* efforts made to preclude measurement outliers, which clearly enhances integrity. However, actually quantifying the integrity, which requires the formulation of a high-confidence upper bound on position error, has not yet been done, to the author's best knowledge, for a vision-based system.

1.6 ***Research Contributions***

The previous sections have motivated the need for developing a framework for image-based navigation integrity and described basic concepts and terms of GPS integrity. This research extends the concept of navigation integrity to image-aided navigation, using GPS integrity definitions and methods as a foundation. The research builds on previous work done in the GPS navigation community and utilizes an existing image-based navigation system construct to ultimately develop an approach to image-based navigation integrity.

The work presented in this dissertation is the first known investigation into image-based integrity, and is pursued with the purpose of establishing a performance baseline for image-based navigation integrity and presenting the development and analysis in terms that could be easily understood by the navigation community. With image-based navigation technology rapidly advancing it is necessary to develop this baseline now. A significant contribution of this dissertation is in initiating a research vein for image-based navigation systems.

Another key contribution of this research is in the rigorous development of a mathematical relationship relating the horizontal position error, which is not directly observable, to an associated test statistic that is outwardly observable in the bearing-type measurement case of the image-based navigation systems described in this research [66]. The new algorithm, described in Chapter IV, produces a value indicative of the horizontal position error and test statistic relationship allowing prediction of the worst case position error, in a deterministic evaluation, relative to a specified threshold defined by user probability of false alarm requirements. Performance evaluations are done in comparison with existing GPS methodologies, considering an angular error in one or potentially both elements of a pixel pair. In the single error case, the results parallel that of GPS methods, but in the dual error case, the results indicate that the new algorithm more accurately captures the impact of the the “worst-case” angular error.

This effort is extended to make an additional contribution in an evaluation of the impact of attitude on the image-based integrity result [65]. Current integrity methods are not known to consider the impact of estimating both position and attitude. The research shows that when extending the image-based navigation problem from one where only position estimation is done to one where both position and attitude are being estimated, there is a significant increase in the magnitude of the potential horizontal position error before possible detection at a defined threshold could occur. The results demonstrate that image-based navigation integrity in image-based navigation potentially gains significant benefit from the use of higher grade inertial measurement units that provide higher accuracy attitude information.

Additional contributions seen in this research include the study and evaluation of the relationship between the image-based integrity performance metric and altitude, the investigation and evaluation of the impact of geometry on image-based navigation integrity, and examination of linearization when estimating position only with respect to the image-based integrity performance metric. Unlike GPS integrity levels, image-based navigation integrity levels can not be readily computed in advance. These additional studies in altitude, geometry, and linearization result in better characterization of algorithm behavior under various conditions, and potentially aid in making future advancements of image-based navigation integrity more predictive of possible performance levels.

1.7 ***Summary***

This chapter provided an introduction to the concepts of navigation integrity, primarily in a GPS context, in order to establish baseline knowledge for the investigation and development of a context for image-based integrity. Chapter II presents conceptual and mathematical background for topics to INS/image-aided navigation and GPS integrity. These topics include least-squares methodology, sensor integration requirements and techniques, imaging theory, and foundational state-of-the-art GPS integrity methods. Chapter III presents simulation results from an investigation of two GPS/INS integrated integrity approaches. A new algorithm for an integrity approach for image-based navigation is developed in Chapter IV. Chapter V presents detailed results and analysis of algorithm performance in multiple scenarios designed to investigate the impact of estimating position and estimating both position and attitude. Finally, Chapter VI provides a summary and conclusion and outlines future research opportunities based on this research.

II. Background

This chapter reviews the mathematical background required to adequately pursue an image-aided navigation integrity algorithm. The chapter first defines common notation used throughout the chapter. The chapter then summarizes least-squares and Kalman filter mathematical derivations, which are both essential to understanding current GPS integrity algorithms and likely future image-aided algorithms. Since the research is based on image-aided navigation and comparison with GPS-aided navigation, the chapter also describes limited basic principles in INS, GPS, and image-based navigation, paying specific attention to the measurement models used in the latter two. The real emphasis in this chapter is in the mathematics behind GPS integrity, using selected foundational concepts spanning from the beginning of GPS integrity to the current state-of-the-art. Finally, a brief synopsis of current image aiding research is explored.

2.1 Mathematical Notation

For clarity, the following describes common mathematical notation throughout this chapter:

- **Scalars:** Scalar quantities are represented by italicized characters (e.g. x or X)
- **Vectors:** Vectors are denoted by bold type characters (e.g., $\mathbf{x}$ or $\boldsymbol{\phi}$) and are composed of scalar elements, normally expressed in column form. For example, x_i indicates the scalar value in the i^{th} row of $\mathbf{x}$.
- **Matrices:** Matrices are denoted by upper case characters in bold font. Reference to a particular element of a matrix is done using row, column indexing (e.g., X_{ij} refers to the element in the i^{th} row and j^{th} column of $\mathbf{X}$).
- **Transpose:** A vector (or matrix) transpose is denoted by a superscript Roman T, as in $\mathbf{x}^T$.

- **Estimated Variables:** Estimates of random variables are denoted by the *hat* character, as in $\hat{\mathbf{x}}$.
- **Computed Variables:** Variables which are corrupted by errors are denoted by the *tilde* character, as in $\tilde{\mathbf{x}}$.
- **Homogeneous Coordinates:** Homogeneous coordinate vectors are defined to have a value of 1 in the last element, and are denoted by an *underline*, as in $\underline{\mathbf{x}}$ [119].
- **Reference Frame:** Navigation vector quantities are defined with respect to a specific coordinate frame, with a superscript letter used to identify the current frame of reference (e.g., $\mathbf{p}^a$ is a vector in the a frame).
- **Direction Cosine Matrices:** Direction cosine matrices are real, unitary, matrices that transform vectors (or matrices) between three dimensional orthogonal coordinate frames (e.g., $\mathbf{C}_a^b$ is the transformation matrix from frame a to frame b).
- **Transformation Matrices:** Transformation matrices are linear operators that map vectors or matrices between $\mathbb{R}^m$ and $\mathbb{R}^n$.
- **Relative Position or Motion:** When necessary to specify relative motion between reference frames, combined subscript letters are used to designate the frames (e.g., $\boldsymbol{\omega}_{ab}^c$ represents the rotation rate vector of the a-frame with respect to the b-frame, expressed in c-frame coordinates) [119].
- **Identity Matrices:** Identity matrices are defined as having the value of 1 for each diagonal entry and 0 everywhere else, and are denoted as $\mathbf{I}$. Where necessary, the dimensions of the identity matrix are provided in subscript notation.
- **Skew-Symmetric Matrices:** A Skew-symmetric matrix for a vector $\mathbf{a} = [a_1, a_2, a_3]^T$, is denoted by $[\mathbf{a}\times]$, and takes the form:

$$[\mathbf{a}\times] = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}$$

2.2 Mathematical Background

2.2.1 ***Least Squares Primer***. As mentioned in Chapter I, the concept of integrity monitoring involves inferring some measure of quality of the position solution from the measurements provided to the system. Vehicle dynamics and sensor measurements are often non-linear. To make the navigation problem tractable, these functions are frequently linearized in order to use well-established mathematical techniques. Indeed, the deterministic measurement model is typically given in linear form as:

$$\mathbf{z} = \mathbf{H}\mathbf{x} \tag{2.1}$$

This simplistic model forms the basis for integrity. The ability to solve this problem in terms of the most accurate value for $\mathbf{x}$ given a potentially noise or error-corrupted value of $\mathbf{z}$ is at the core of integrity algorithms. Integrity also requires measurement redundancy, i.e., $\mathbf{H}$ is overdetermined, with m equations to solve for n unknowns ($m > n$). Since the GPS-only approaches to integrity began with the least-squares method, it is worth quickly summarizing here. Except where noted, the discussion follows the presentations by [3] and [108].

The fact that the observation matrix, $\mathbf{H}$ is overdetermined means that the the solution to $\mathbf{z} = \mathbf{H}\mathbf{x}$ will likely be inconsistent, resulting in $\mathbf{z} - \mathbf{H}\mathbf{x} \neq \mathbf{0}$. The least-squares approach is to find an estimate of $\mathbf{x}$, denoted as $\hat{\mathbf{x}}$, which minimizes the value of $||\mathbf{z} - \mathbf{H}\mathbf{x}||$ with respect to the Euclidian inner product. Letting *U* be the column space of $\mathbf{H}$, the problem is best described geometrically as finding the $\hat{\mathbf{x}}$ in $\mathbb{R}^n$ such that $\mathbf{H}\hat{\mathbf{x}}$ is the vector in *U* that is closest to $\mathbf{z}$. This closest vector is the orthogonal projection of $\mathbf{z}$ onto *U*, and since *U* lies in the column space of $\mathbf{H}$, then $||\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}||$ lies in the nullspace of $\mathbf{H}^T$. As a result, the least squares solution is given by:

$$\mathbf{H}^T(\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}) = 0 \tag{2.2}$$

or, by multiplying through and rearranging terms:

$$\mathbf{H}^T\mathbf{H}\hat{\mathbf{x}} = \mathbf{H}\mathbf{z} \tag{2.3}$$

which is also called the normal equation in statistics [108]. If the columns of $\mathbf{H}$ are linearly independent, then $\mathbf{H}^T\mathbf{H}$ is invertible, and the best estimate of $\hat{\mathbf{x}}$ can be given as:

$$\hat{\mathbf{x}} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{z} \tag{2.4}$$

The above analysis assumes that each measurement is of equal value. The case often arises where some observations have higher merit than others, and thus should have greater influence on the estimated solution. Therefore, it is sometimes valuable to apply weighting to the measurements, which can be done by multiplying both sides of the original measurement equation by a full rank weighted matrix, $\mathbf{W}$:

$$\mathbf{W}\mathbf{H}\mathbf{x} = \mathbf{W}\mathbf{z} \tag{2.5}$$

such that weighted (or generalized) least squares solution can be expressed:

$$\mathbf{H}^T\mathbf{W}^T\mathbf{W}\mathbf{H}\hat{\mathbf{x}} = \mathbf{H}^T\mathbf{W}^T\mathbf{W}\mathbf{z} \tag{2.6}$$

or letting $\mathbf{C} = \mathbf{W}^T\mathbf{W}$:

$$\mathbf{H}^T\mathbf{C}\mathbf{H}\hat{\mathbf{x}} = \mathbf{H}^T\mathbf{C}\mathbf{z} \tag{2.7}$$

such that the weighted estimate is given by:

$$\hat{\mathbf{x}}_W = (\mathbf{H}^T\mathbf{C}\mathbf{H})^{-1}\mathbf{H}^T\mathbf{C}\mathbf{z} \tag{2.8}$$

Typically, the best choice for an unbiased matrix $\mathbf{C}$ is the inverse covariance matrix, whose diagonal entries are the variances of the error for each measurement, and whose off-diagonal terms are the cross-covariances of the measurement errors [108]. This formulation is convenient when extending to the case where the measurement model is influenced by zero-mean, additive white Gaussian noise (AWGN), v, e.g. $\mathbf{z} = \mathbf{H}\mathbf{x} + \mathsf{v}$, whose covariance matrix is commonly represented as the diagonal matrix $\mathbf{R}$. Thus:

$$\hat{\mathbf{x}}_W = (\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})^{-1}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{z} \tag{2.9}$$

Under the condition that the $\mathbf{R}$ matrix is the diagonal matrix of the error covariances, this model is considered a minimum-variance estimator, and also the “best linear unbiased estimator,” according to the Gauss-Markov theorem [62]. In addition, if v is normally distributed with a mean of zero and covariance $\mathbf{R}$, Equation (2.9) is the maximum likelihood estimator, of which a Kalman filter is the recursive equivalent [62], when the conditions that the state vector is a random constant, $\mathbf{H}$ is overdetermined, and there is no *a priori* knowledge about $\mathbf{x}$ hold [19]. The covariance of the optimal estimate, $\hat{\mathbf{x}}$, is given by [48; 62]:

$$\mathbf{P}_{\hat{\mathbf{x}}\hat{\mathbf{x}}} = (\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})^{-1} \tag{2.10}$$

with the expected value:

$$E\{\hat{\mathbf{x}}\} = \mathbf{x} \tag{2.11}$$

This discussion is meant to serve as a primer for the following sections. It is easy to see the relationships between the observation matrix, the measurement vector and the least-squares estimate vector from this simple illustration. It is also easy to visualize that if $\hat{\mathbf{x}}$ is the best single epoch estimate of a navigation position solution (or more precisely, the position errors in an error state model), how perturbations in $\mathbf{z}$ or $\mathbf{H}$ would affect the quality of the estimate. In part, the problem of integrity is determining when these perturbations have compromised the navigation solution.

2.2.2 ***The Kalman Filter.*** The Kalman filter can be used to handle modeling, estimation, and control, producing meaningful design and analysis results. Under the conditions given in the previous section, the filter operates as a recursive least-squares filter. However, this filter serves to replace the least-squares approach when the state vector is stochastic with behavior that varies as a function of time [48]. The Kalman filter is efficient in these cases, since it does not require the storage of all previous data for the incorporation of a new measurement [80]. Instead, the filter propagates forward the best estimate of the state variables of interest, starting with the *a priori* information, and forms a new best estimate based on the propagated estimate and the incoming measurement. The process repeats, continually refining the estimate of the variables under consideration.

Under the assumptions that the system being modeled is linear, and that the system and measurement noises are white and exhibit Gaussian statistics, the Kalman filter is an optimal estimator [80]. The ability of the Kalman filter to synthesize all available measurements, including noise-corrupted ones, makes it an especially powerful technique in modeling and estimating stochastic processes. The filter may yield suboptimal results if the conditions of the physical system or measurement devices that form the basis of the filter models were to change. This characteristic often aids in fault detection, provided the system does not become corrupted by these changes. In situations in which models are not linear, it may be possible to utilize the linearized Kalman filter or extended Kalman filter discussed in later sections. Mathematical descriptions of the filter models follow [80] and [81].

2.2.3 ***State and Measurement Model Equations.*** Physical systems are typically described by continuous-time dynamics. For the purposes of computer simulation and analysis, it is often desirable to convert the dynamics model to discrete-time as well. Both the continuous-time and discrete-time representations are presented below. The sampled-data measurements are inherently discrete-time.

A linear continuous-time state model can be expressed in state-space form [80]:

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) + \mathbf{G}(t)\mathbf{w}(t) \tag{2.12}$$

where

$\mathbf{x}(t)$ = the $(n \times 1)$ system state vector

$\mathbf{F}(t)$ = the $(n \times n)$ state dynamics matrix

$\mathbf{B}(t)$ = the $(n \times r)$ control input matrix

$\mathbf{u}(t)$ = the $(r \times 1)$ deterministic control input vector

$\mathbf{G}(t)$ = the $(n \times s)$ noise input matrix

$\mathbf{w}(t)$ = $(s \times 1)$ dynamics driving noise vector

where $\mathbf{w}(t)$ is a zero-mean white Gaussian noise of strength $\mathbf{Q}(t)$:

$$E\{\mathbf{w}(t)\mathbf{w}(t+\tau)\} = \mathbf{Q}(t)\delta(\tau) \tag{2.13}$$

and $\delta(\bullet)$ is the Dirac delta function.

At discrete times the solution to Equation (2.12) may be written as:

$$\begin{aligned}\mathbf{x}(t_k) = \mathbf{\Phi}(t_k, t_{k-1})\mathbf{x}(t_{k-1}) &+ \left[\int_{t_{k-1}}^{t_k} \mathbf{\Phi}(t_k, \tau)(\tau)d\tau\right]\mathbf{u}(t_{k-1}) \\ &+ \left[\int_{t_{k-1}}^{t_k} \mathbf{\Phi}(t_k, \tau)\mathbf{G}(\tau)\boldsymbol{d\beta}(\tau)\right]\end{aligned} \tag{2.14}$$

where it has been assumed that $\mathbf{u}(t)$ is maintained at a constant vector over the interval from t_k to t_{k-1} (as would be the case if it were generated as the output of a digital computer put through a zero-order hold).

The equivalent discrete-time stochastic difference equation can then be expressed:

$$\mathbf{x}(t_k) = \mathbf{\Phi}(t_k, t_{k-1})\mathbf{x}(t_{k-1}) + \mathbf{B_d}(t_{k-1})\mathbf{u}(t_{k-1}) + \mathbf{G_d}(t_{k-1})\mathbf{w_d}(t_{k-1}) \tag{2.15}$$

The state transition matrix, $\mathbf{\Phi}(t_k, t_{k-1})$, is found as a solution to:

$$\frac{d}{dt}\mathbf{\Phi}(t, t_{k-1}) = \mathbf{F}(t)\mathbf{\Phi}(t, t_{k-1}),\ \mathbf{\Phi}(t_{k-1}, t_{k-1}) = \mathbf{I} \tag{2.16}$$

integrated from t_{k-1} to t_k. In the special case where $\mathbf{F}(t)$ is constant, the state transition matrix may be defined as:

$$\mathbf{\Phi}(t_k, t_{k-1}) = \mathbf{\Phi}(\Delta t) = e^{\mathbf{F}\Delta t} \tag{2.17}$$

where $\Delta t \triangleq t_k - t_{k-1}$

The discrete-time input matrix and discrete-time noise vector are determined from:

$$\mathbf{B}_{\mathrm{d}}(t_{k-1}) = \int_{t_{k-1}}^{t_k} \mathbf{\Phi}(t_k, \tau)\mathbf{B}(\tau)d\tau \tag{2.18}$$

$$\mathbf{w}_{\mathrm{d}}(t_{k-1}) = \int_{t_{k-1}}^{t_k} \mathbf{\Phi}(t_k, \tau)\mathbf{G}(\tau)\boldsymbol{d\beta}(\tau) \tag{2.19}$$

with $\mathbf{G}_{\mathrm{d}}(t_{k-1})$ assumed to be the identity matrix $\mathbf{I}$. The discrete-time white Gaussian dynamics driving noise has the statistics:

$$E\{\mathbf{w}_{\mathrm{d}}(t_k)\} = \mathbf{0} \tag{2.20}$$

$$E\{\mathbf{w}_{\mathrm{d}}(t_k)\mathbf{w}_{\mathrm{d}}^T(t_k)\} = \mathbf{Q}_{\mathrm{d}}(t_k) = \int_{t_{k-1}}^{t_k} \mathbf{\Phi}(t_k, \tau)\mathbf{G}(\tau)\mathbf{Q}(\tau)\mathbf{G}^T(\tau)\mathbf{\Phi}^T(t_k, \tau)d\tau \tag{2.21}$$

$$E\{\mathbf{w}_{\mathrm{d}}(t_k)\mathbf{w}_{\mathrm{d}}^T(t_j)\} = \mathbf{0},\ \mathrm{t}_k \neq \mathrm{t}_j \tag{2.22}$$

As mentioned previously, the measurements given to the filter are typically sampled sensor data. The measurement model is then naturally expressed in discrete-time form:

$$\mathbf{z}(t_k) = \mathbf{H}(t_k)\mathbf{x}(t_k) + \mathbf{v}(t_k) \tag{2.23}$$

The statistics of the associated measurement corruption noise, $\mathbf{v}(t_k)$, are given by:

$$E\{\mathbf{v}(t_k)\} = \mathbf{0} \tag{2.24}$$

$$E\{\mathbf{v}(t_k)\mathbf{v}^T(t_j)\} = \mathbf{R}(t_k)\delta_{jk} \tag{2.25}$$

where δ_{jk} is the Kronecker delta. The independence of the dynamics driving noise $\mathbf{w}_\mathrm{d}(t_k)$ and the measurement corruption noise $\mathbf{v}(t_k)$ is assumed, resulting in:

$$E\{\mathbf{w}_\mathrm{d}(t_k)\mathbf{v}^T(t_j)\} = \mathbf{0} \text{ for all } k \text{ and } j \tag{2.26}$$

*2.2.4 **Kalman Filter Propagation and Update**.* The Kalman filter propagates forward in time from t_{k-1} to t_k (or equivalently from t_k to t_{k+1}) starting from the last update cycle state and covariance estimates. Any available *a priori* information about the initial state and covariance estimates is used in the first propagation cycle. The superscript "$-$" is used to denote the propagated estimate immediately before update at a given sample time, while the "$+$" indicates the result of the update after measurement incorporation.

$$\hat{\mathbf{x}}(t_k^-) = \mathbf{\Phi}(t_k,t_{k-1})\hat{\mathbf{x}}(t_{k-1}^+) + \mathbf{B}_\mathrm{d}(t_{k-1})\mathbf{u}(t_{k-1}) \tag{2.27}$$

$$\mathbf{P}(t_k^-) = \mathbf{\Phi}(t_k,t_{k-1})\mathbf{P}(t_{k-1}^+)\mathbf{\Phi}^T(t_k,t_{k-1}) + \mathbf{G}_\mathrm{d}(t_{k-1})\mathbf{Q}_\mathrm{d}(t_{k-1})\mathbf{G}_\mathrm{d}^T(t_{k-1}) \tag{2.28}$$

When measurements become available, the Kalman filter is updated by:

$$\mathbf{K}(t_k) = \mathbf{P}(t_k^-)\mathbf{H}^T(t_k)\left[\mathbf{H}(t_k)\mathbf{P}(t_k^-)\mathbf{H}^T(t_k) + \mathbf{R}(t_k)\right]^{-1} \tag{2.29}$$

$$\hat{\mathbf{x}}(t_k^+) = \hat{\mathbf{x}}(t_k^-) + \mathbf{K}(t_k)\left[\mathbf{z}_i - \mathbf{H}(t_k)\hat{\mathbf{x}}(t_k^-)\right] \tag{2.30}$$

$$\mathbf{P}(t_k^+) = \mathbf{P}(t_k^-) - \mathbf{K}(t_k)\mathbf{H}(t_k)\mathbf{P}(t_k^-) \tag{2.31}$$

The residuals in Equation (2.30) are the computed differences between the incoming measurement and the filter's best prediction of what the measurement would be. The Kalman filter residual vector is defined as:

$$\mathbf{r}(t_k) = \mathbf{z}(t_k) - \mathbf{H}(t_k)\hat{\mathbf{x}}(t_k^-) \tag{2.32}$$

with a filter-computed covariance:

$$\mathbf{A}(t_k) = \mathbf{H}(t_k)\mathbf{P}(t_k^-)\mathbf{H}^T(t_k) + \mathbf{R}(t_k) \tag{2.33}$$

The residuals and their associated covariance provide for the monitoring of filter performance, and provide indications as to whether the assumptions of adequacy of the filter model are being satisfied. Under the proper conditions, a well-modeled filter should exhibit residuals which are zero-mean, white, and Gaussian in nature, with the expected covariance $\mathbf{A}(t_k)$.

2.3 The Linearized and Extended Kalman Filters

In problems in which nonlinearities in the deterministic portion of the dynamics model or measurement model are not truly negligible [81], it may be possible to use extensions of the linear Kalman filter. The development of the linearized and extended Kalman filters allow for such cases, provided additional assumptions are met. For example, the previously stated noise statistics are still required, but are now also assumed to be additive in a linear fashion, and the nonlinear dynamics model must meet specific criteria for continuity described in [81]. By using the linearized and extended Kalman filter techniques to approximate a true optimal nonlinear filter, many nonlinear problems of interest can still be addressed using linear estimation theory.

In linearized Kalman filtering, the state and measurement equations are linearized about a nominal state trajectory in order to estimate small changes in the states with respect to that trajectory. If the time interval over which the changes occur is kept sufficiently small, these deviations may be evaluated as linear perturbations over the interval. Adding this optimal estimate of the change in the state to the nominal state value produces an overall estimate of the total state. This technique has specific application in GPS/INS integration schemes shown later in this chapter.

The extended Kalman filter relinearizes about the the most recent total state estimate of the trajectory, instead of using the fixed nominal trajectory [125]. By relinearizing about each new state estimate as it is computed, the deviations relative to the newly declared nominal trajectory are expected to remain sufficiently small, further validating the use of linear perturbation techniques.

2.3.1 ***State and Measurement Model Equations.***

Let the system be well modeled by the following nonlinear, stochastic differential equation:

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), t] + \mathbf{G}(t)\mathbf{w}(t) \tag{2.34}$$

where $\mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), t]$ is described as a nonlinear function of three arguments: the n-dimensional state vector $\mathbf{x}(t)$, the r-dimensional deterministic control input function $\mathbf{u}(t)$, and of continuous-time t. The dynamics driving noise vector, $\mathbf{w}(t)$, is a zero-mean, white Gaussian noise process with covariance kernel:

$$E\{\mathbf{w}(t)\mathbf{w}^T(t+\tau)\} = \mathbf{Q}(t)\delta(\tau) \tag{2.35}$$

as described in preceding sections, with τ having units of time.

The nonlinear discrete-time measurement equation, modeled as a function of state and time, plus linearly additive white noise, takes the form:

$$\mathbf{z}(t_k) = \mathbf{h}[\mathbf{x}(t_k), t_k] + \mathbf{v}(t_k) \tag{2.36}$$

where $\mathbf{z}(t_k)$ is the m-dimensional measurement vector and $\mathbf{v}(t_k)$ is the m-dimensional discrete-time measurement corruption white noise vector with the following statistics:

$$E\{\mathbf{v}(t_k)\} = \mathbf{0} \tag{2.37}$$

$$E\{\mathbf{v}(t_k)\mathbf{v}^T(t_j)\} = \begin{cases} \mathbf{R}(t_k) & \text{for } t_k = t_j \\ \mathbf{0} & \text{for } t_k \neq t_j \end{cases} \tag{2.38}$$

*2.3.2 **The Linearized Kalman Filter**.* The linearized Kalman filter is based on perturbation states about the nominal state trajectory, $\mathbf{x}_n(t)$, which satisfy both the initial condition $\mathbf{x}_n(t_0) = \mathbf{x}_{n0}$ and the deterministic differential equation [125]:

$$\dot{\mathbf{x}}_n(t) = \mathbf{f}[\mathbf{x}_n(t), \mathbf{u}(t), t] \tag{2.39}$$

The associated measurements, also defined by the nominal trajectory, are defined as:

$$\mathbf{z}_n(t_k) = \mathbf{h}[\mathbf{x}_n(t_k), t_k] \tag{2.40}$$

The perturbation of the state is determined by differencing the original states, Equation (2.34) and the nominal states, Equation (2.39), which yields a stochastic process satisfying [81]:

$$\delta\dot{\mathbf{x}}(t) = [\dot{\mathbf{x}}(t) - \dot{\mathbf{x}}_n(t)] = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), t] - \mathbf{f}[\mathbf{x}_n(t), \mathbf{u}(t), t] + \mathbf{G}(t)\mathbf{w}(t) \tag{2.41}$$

Approximating $\delta\dot{\mathbf{x}}(t) = [\dot{\mathbf{x}}(t) - \dot{\mathbf{x}}_n(t)]$ to first order through a truncated Taylor series expansion, the perturbation equation can be written as:

$$\delta\dot{\mathbf{x}}(t) = \mathbf{F}[t; \mathbf{x}_n(t)]\delta\mathbf{x}(t) + \mathbf{G}(t)\mathbf{w}(t) \tag{2.42}$$

where $\mathbf{F}[t; \mathbf{x}_n(t)]$ is the linearized dynamics matrix (n-by-n), found by taking the partial derivatives of $\mathbf{f}\left[\mathbf{x}(t), \mathbf{u}(t), t\right]$ with respect to $\mathbf{x}(t)$, and evaluating them along the nominal trajectory, $\mathbf{x}_n(t)$ [125]:

$$\mathbf{F}[t; \mathbf{x}_n(t)] \triangleq \left.\frac{\partial\mathbf{f}[\mathbf{x}, \mathbf{u}(t), t]}{\partial\mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_n(t)} \tag{2.43}$$

The derivation of the discrete-time perturbation measurements parallels the previous discussion, resulting in the difference equation being described as:

$$\delta\mathbf{z}(t_k) = [\mathbf{z}(t_k) - \mathbf{z}_n(t_k)] = \mathbf{h}[\mathbf{x}(t_k), t_k] - \mathbf{h}[\mathbf{x}_n(t_k), t_k] + \mathbf{v}(t_k) \tag{2.44}$$

with the first order approximation of $\delta\mathbf{z}(t_k)$:

$$\delta\mathbf{z}(t_k) = \mathbf{H}[t_k; \mathbf{x}_n(t_k)]\delta\mathbf{x}(t_k) + \mathbf{v}(t_k) \tag{2.45}$$

where $\mathbf{H}[t_k; \mathbf{x}_n(t_k)]$ is the linearized observation matrix (m-by-n), found by taking the partial derivatives of $\mathbf{h}\left[\mathbf{x}(t_k), t_k\right]$ with respect to $\mathbf{x}(t_k)$, and again evaluating them along the nominal trajectory:

$$\mathbf{H}[t_k; \mathbf{x}_n(t_k)] \triangleq \left.\frac{\partial\mathbf{h}[\mathbf{x}, t_k]}{\partial\mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_n(t_k)} \tag{2.46}$$

The linearized Kalman filter produces an estimate of $\delta\mathbf{x}(t)$, denoted as $\delta\hat{\mathbf{x}}(t)$, which is used to form the best estimate of the total state:

$$\hat{\mathbf{x}}(t) \triangleq \mathbf{x}_n(t) + \delta\hat{\mathbf{x}}(t) \tag{2.47}$$

Linear filter theory may now be applied to Equations (2.42) and (2.45), provided the required partial derivatives exist. Filter performance is conditioned on the perturbations remaining small enough that approximations made by truncating the Taylor series expansion to first-order are justified. Compensatory adjustments of the white noise strengths of $\mathbf{w}(t)$ in Equation (2.42) and/or the covariance of the noise $\mathbf{v}(t_k)$ in Equation (2.45) should be considered in an effort to account for the loss of higher order terms.

2.3.3 ***Extended Kalman Filter Equations***. In extended Kalman filtering, the filter is relinearized about the new estimate of the total state, $\hat{\mathbf{x}}(t_k^+)$, once it is formed, enhancing the assumptions made when using linear perturbation techniques by incorporating this new reference trajectory into the estimation process. Starting from the initial conditions, $\hat{\mathbf{x}}(t_{k-1}/t_{k-1}) = \hat{\mathbf{x}}(t_{k-1}^+)$ and $\mathbf{P}(t_{k-1}/t_{k-1}) = \mathbf{P}(t_{k-1}^+)$ (using $\hat{\mathbf{x}}_0$ and $\mathbf{P}_0$ for the first cycle), the extended Kalman filter propagates the estimate forward to the next sample t_k time by integrating [81]:

$$\dot{\hat{\mathbf{x}}} = \mathbf{f}[\hat{\mathbf{x}}(t/t_{k-1}), \mathbf{u}(t), t] \tag{2.48}$$

$$\dot{\mathbf{P}}(t/t_{k-1}) = \mathbf{F}[t; \hat{\mathbf{x}}(t/t_{k-1})]\mathbf{P}(t/t_{k-1}) + \mathbf{P}(t/t_{k-1})\mathbf{F}^T[t; \hat{\mathbf{x}}(t/t_{k-1})] + \mathbf{G}(t)\mathbf{Q}(t)\mathbf{G}^T(t) \tag{2.49}$$

where the time argument t/t_{k-1} denotes the realization of the variable at time t, conditioned on the measurements through time t_{k-1} [125]. $\mathbf{F}[t; \hat{\mathbf{x}}(t/t_{k-1})]$ is the n-by-n partial derivative matrix linearized about the state estimate $\hat{\mathbf{x}}(t/t_{k-1})$:

$$\mathbf{F}[t;\hat{\mathbf{x}}(t/t_{k-1})] \triangleq \left.\frac{\partial \mathbf{f}[\mathbf{x},\mathbf{u}(t),t]}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}(t/t_{k-1})} \tag{2.50}$$

for all t in the interval $[t_{k-1}, t_k)$. After integrating Equations (2.48) and (2.49) to the next sample time, $\hat{\mathbf{x}}(t_k^-)$ and $\mathbf{P}(t_k^-)$ are defined as:

$$\hat{\mathbf{x}}(t_k^-) = \hat{\mathbf{x}}(t_\mathrm{i}/t_{\mathrm{i}-1}) \tag{2.51}$$

$$\mathbf{P}(t_k^-) = \mathbf{P}(t_\mathrm{i}/t_{\mathrm{i}-1}) \tag{2.52}$$

The extended Kalman filter incorporates the measurement $\mathbf{z}(t_k) = \mathbf{z}_i$ by means of:

$$\mathbf{K}(t_k) = \mathbf{P}(t_k^-)\mathbf{H}^T[t_k;\hat{\mathbf{x}}(t_k^-)]\{\mathbf{H}[t_k;\hat{\mathbf{x}}(t_k^-)]\mathbf{P}(t_k^-)\mathbf{H}^T[t_k;\hat{\mathbf{x}}(t_k^-)] + \mathbf{R}(t_k)\}^{-1} \tag{2.53}$$

$$\hat{\mathbf{x}}(t_i^+) = \hat{\mathbf{x}}(t_i^-) + \mathbf{K}(t_k)\{\mathbf{z}_i - \mathbf{h}[\hat{\mathbf{x}}(t_k^-), t_k]\} \tag{2.54}$$

$$\mathbf{P}(t_i^+) = \mathbf{P}(t_i^-) - \mathbf{K}(t_k)\mathbf{H}[t_k;\hat{\mathbf{x}}(t_k^-)]\mathbf{P}(t_i^-) \tag{2.55}$$

where $\mathbf{H}[t_k;\mathbf{x}(t_k^-)]$ is the m-by-n observation matrix, linearized about the state estimate:

$$\mathbf{H}[t_k;\hat{\mathbf{x}}(t_k^-)] \triangleq \left.\frac{\partial \mathbf{h}[\mathbf{x},t_k]}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}(t_k^-)} \tag{2.56}$$

2.4 *Strapdown Inertial Navigation System*

Inertial navigation is the method of determining a vehicle's current position based on the vehicle's point of origin, how long it has traveled, and the velocity and direction of movement based upon measurements of specific force and rotation. An inertial navigation system (INS) is a mechanized way of performing this navigation [112]. Modern strapdown inertial navigation systems are composed of an inertial measurement unit (IMU), digital computers, and navigation algorithms [112]. A standard IMU for three dimensional navigation contains three accelerometers to measure specific force, $(\mathbf{f})$, and three gyroscopes to measure angular rate of change ($\boldsymbol{\omega}$) [112]. The orientation of the accelerometers and gyros is fixed with respect to the vehicle body frame in the strapdown INS.

Gyroscopes measure the rate of change in the vehicle's orientation with respect to the inertial reference frame [119]. This relationship is denoted $\boldsymbol{\omega}_{ib}^{b}$, where the superscript indicates the observation frame ($\underline{\text{b}}$ody or b-frame here), and the subscript indicates frame of reference frame and the measurement frame, respectively (here, from the $\underline{\text{i}}$nertial or i-frame to the b-frame). Integrating the angular rate values forward from the initial conditions yields the vehicle's attitude angles. The attitude computer within the INS uses the angular rate values from these sensors to transform the specific force from the vehicle body frame into the inertial reference frame. Transformations into other coordinate frames are often desired, and can be accomplished using direction cosine matrices related to the specific frame of interest [45].

Potential reference frames include the Earth-fixed inertial frame (the previously stated i-frame), Earth-centered, Earth-fixed frame (ECEF or e-frame), the vehicle body frame (the previously stated b-frame), or a local navigation frame (n-frame), of which North-East-Down (NED) and Latitude, Longitude, and Altitude (LLh, denoted as $L\lambda h$ to delineate Latitude and Longitude) are two examples. An additional frame of reference brought on by image-aiding is the camera frame (c-frame), which will be discussed later. Some examples of alternative reference frames are shown in Figure 2.1, namely the NED frame and the ECEF frame. The following discussions will primarily focus on a local navigation frame of reference. For detailed theory and model development, the work by [13; 100; 112; 119] provide excellent descriptions of reference frames and associated INS dynamics models.

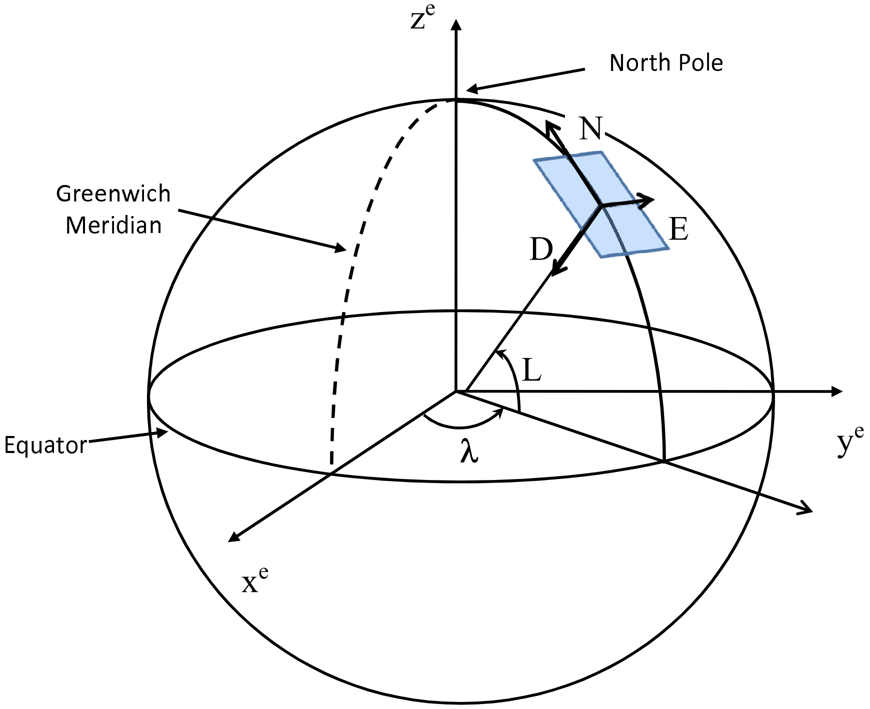

Figure 2.1: Examples of Navigation Reference Frames.

Specific force measured by an accelerometer is not exclusively the acceleration of the vehicle; it is the sum of the acceleration with respect the the inertial frame and acceleration due to gravity. The INS must include a gravity model and the Earth's rate, $\boldsymbol{\omega}_{ie}^{i}$ or $\boldsymbol{\omega}_{ie}^{e}$, in its navigation computer to separate acceleration from the specific force measurements. Acceleration is the first derivative of velocity and the second derivative of position. Given the appropriate initial conditions and the acceleration value, resolved through the above method, the vehicle velocity and position can be determined through integration.

Typical strapdown implementations use the IMU to generate angular rate and specific force measurements as discrete values proportional to the sensed motion. These values are identified as $\Delta\Theta$ (delta-theta) and Δv (delta-v) pulses (shown in Figure 2.2), respectively. Using a digital computer, the "integration" is actually done by accumulating the $\Delta\Theta$'s and Δv's over time. This computer processing is a function of the inertial navigation unit (INU), a unit primarily made up of navigation and gravity computers used to

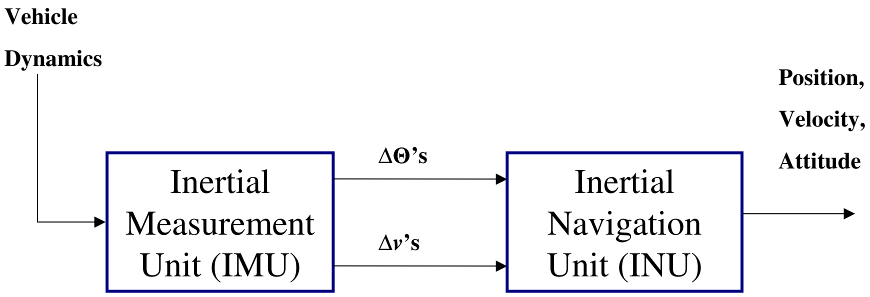

Figure 2.2: Inertial Navigation System Block Diagram.

develop a navigation solution based on navigation equations (like those presented in the following discussion).

The accelerometers and gyroscopes report measurements in the body frame, but the navigation equations may be solved in a number of different reference frames. Choosing a specific reference frame, or mechanization scheme, for implementation is typically application dependent. Different mechanization schemes must account for differences in the way "apparent forces," such as Coriolis and centripetal forces, are modeled. The following material presents a basic summary of the navigation equations associated with the *n*-frame mechanization.

Since most navigation is performed over long distances on the Earth [119], the INS mechanization in a local level navigation frame mechanization is often convenient. The navigation equations for this mechanization are given in terms of the acceleration, the derivative of velocity, $\dot{\mathbf{v}}$, as [13; 112]:

$$\begin{aligned} \dot{\mathbf{v}}_e^n &= \mathbf{C}_b^n \mathbf{f}^b - (2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) \times \mathbf{v}_e^n + \mathbf{g}_l^n \\ &= \mathbf{C}_b^n \mathbf{f}^b - (\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{in}^n) \times \mathbf{v}_e^n + \mathbf{g}_l^n \end{aligned} \tag{2.57}$$

where

$\mathbf{C}_b^n$ is the direction cosine matrix used to transform a vector from the body frame to the navigation frame

$\boldsymbol{\omega}_{ie}^n$ is the angular rate of the Earth frame with respect to the inertial frame, as seen from the navigation frame

$\boldsymbol{\omega}_{en}^n$ is the angular rate of the navigation frame with respect to the Earth frame, as seen from the navigation frame (i.e., transport rate)

$\boldsymbol{\omega}_{in}^n$ is the angular rate of the Earth frame with respect to the inertial frame, as seen from the navigation frame

$\mathbf{g}_l^n$ is the local gravity vector expressed in the navigation frame

Let $\boldsymbol{\Omega}_{nb}^b$ be the skew-symmetric matrix formed from the elements of $\boldsymbol{\omega}_{nb}^b = [\omega_x, \omega_y, \omega_z]^T$, where ω_x, ω_y, and ω_z represent angular rate measurements of the body frame with respect to the navigation frame, as observed from the body frame. $\boldsymbol{\Omega}_{nb}^b$ is then defined as [112]:

$$\boldsymbol{\Omega}_{nb}^b = \boldsymbol{\omega}_{nb}^b \times = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{2.58}$$

where $\boldsymbol{\omega}_{nb}^b$ is computed as:

$$\begin{aligned} \boldsymbol{\omega}_{nb}^b &= \boldsymbol{\omega}_{ib}^b - \mathbf{C}_n^b \boldsymbol{\omega}_{in}^n \\ &= \boldsymbol{\omega}_{ib}^b - (\mathbf{C}_b^n)^T (\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) \end{aligned} \tag{2.59}$$

with $\boldsymbol{\omega}_{ib}^b = [p, q, r]^T$ where p,q, and r represent angular rate measurements provided by the gyroscopes in the three axes. The direction cosine matrix, $\mathbf{C}_b^n$, is propagated forward using [112]:

$$\dot{\mathbf{C}}_b^n = \mathbf{C}_b^n \boldsymbol{\Omega}_{nb}^b \tag{2.60}$$

Figure 2.3 shows the navigation frame mechanization.

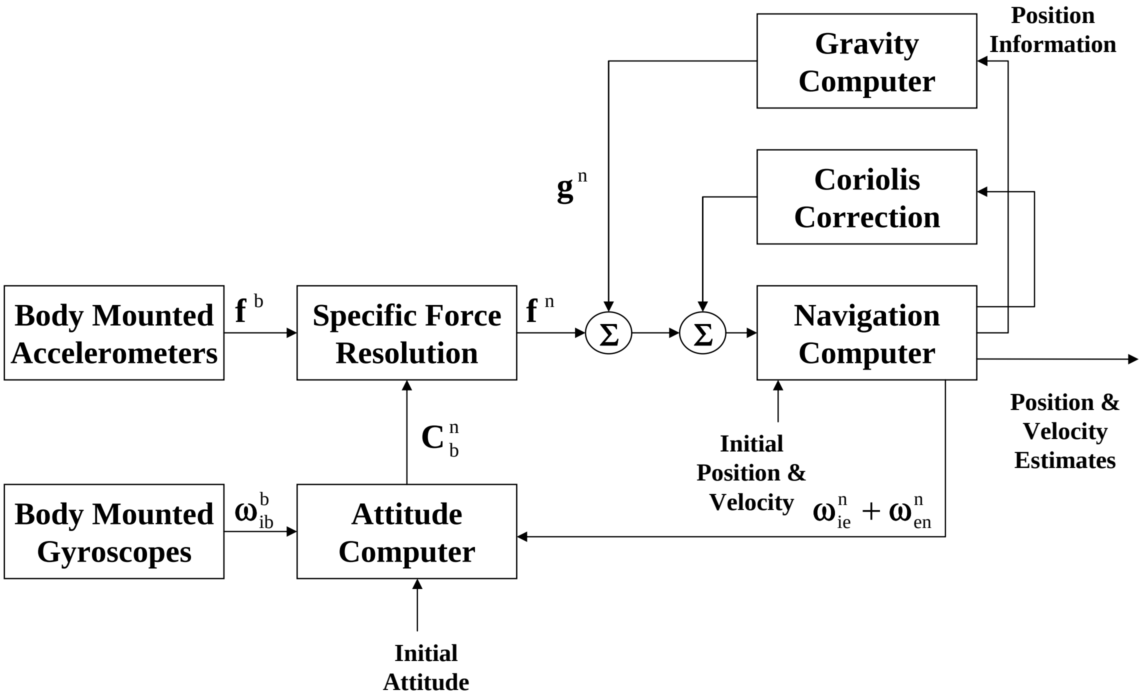

Figure 2.3: Block Diagram Illustrating the Navigation Frame Mechanization of Strapdown INS (based on [112]).

Expressing the vector quantities described above, in terms of the individual components, the navigation equations are [13; 112]:

$$\dot{v}_N = f_N - v_E(2\Omega + \dot{\lambda})\sin(L) + v_D\dot{L} + \eta g \tag{2.61}$$

$$\dot{v}_E = f_E + v_N(2\Omega + \dot{\lambda})\sin(L) + v_D(2\Omega + \dot{L})\cos(L) - \xi g \tag{2.62}$$

$$\dot{v}_D = f_D + v_E(2\Omega + \dot{\lambda})\cos(L) - v_N\dot{L} + g \tag{2.63}$$

where

f_N, f_E, f_D are the specific forces in the North, East and Down directions

v_N, v_E, v_D are the velocity components in the North, East and Down directions

L is the current latitude

Ω is the Earth's rotation rate, 7.292115×10^{-5} rad/s

η, ξ are the angular deflections in the direction of the normal gravity vector observed in the North and East axes, respectively, due to variations in Earth's gravitational field

g is the vertical gravity vector due to the mass attraction of Earth

$\dot{L}$ is the time rate of change of the latitude

$\dot{\lambda}$ is the time rate of change of the longitude

Accounting for the fact that the Earth is ellipsoidal, the time rates of change for latitude, longitude and height above the surface of the Earth are given by [112]:

$$\dot{L} = \frac{v_N}{(R_N + h)} \tag{2.64}$$

$$\dot{\lambda} = \frac{v_E \sec(L)}{(R_E + h)} \tag{2.65}$$

$$\dot{h} = -v_D \tag{2.66}$$

where

$$h = \text{height above the Earth}$$

$$R_N = \text{meridian radius of curvature} = \frac{R(1 - e^2)}{(1 - e^2 \sin^2(L))^{3/2}} \tag{2.67}$$

$$R_E = \text{transverse radius of curvature} = \frac{R}{(1 - e^2 \sin^2(L))^{1/2}} \tag{2.68}$$

with R equal to the length of Earth's semi-major axis and e representing the major eccentricity of the ellipsoid.

It should be noted that the $\sec(L)$ term in Equation (2.65) is singular at $\pm 90^\circ$ latitude, which is problematic when navigating around Earth's poles. The wander azimuth axis system may be used to avoid this singularity [112], and should be considered if navigating at higher latitudes. Details of the wander azimuth frame will not be presented here, but may be found in [13; 100; 112].

In concluding this section, it is important to identify a couple of key characteristics of the strapdown INS. The INS is completely self-contained, using its own inertially-sensed and derived information to render a navigation solution [112]. Many aerospace applications require this type of system independence. INS also exhibits superior high frequency response [13; 45; 50]. This attribute reflects good performance in high vehicle dynamics situations. However, errors resulting from imperfect initial conditions and gravity computations, limit INS accuracy. Significant potential error sources of the INS result from physical limitations of the accelerometers and gyroscopes, particularly gyro drift rates. These errors propagate with time, resulting in poor long-term position estimation and degraded low-frequency response [13; 45; 50]. To attain better long-term performance, the INS system is usually aided with other systems, as was stated in Chapter I, using GPS as an example, with the INS providing a reference trajectory for an integration filter, like the linearized or extended Kalman filters.

2.5 Global Positioning System

The Global Positioning System (GPS) is a space-based navigation system providing accurate three-dimensional velocity and position information to users worldwide [60]. The system employs a minimum of 24 satellites in medium earth orbit to guarantee continuous coverage [84]. GPS navigation provides very good long-term accuracy, but is adversely affected by high frequency error in short-term performance, while INS has complementary attributes, having good short-term accuracy, but being prone to low-frequency drift caused by sensor errors [48]. Aiding of the GPS receiver with other sensors, particularly an INS, may loosen the restrictions on the number of required satellites and improves the overall navigation solution in high dynamics environments.

GPS utilizes the concept of signal time-of-arrival and operates passively, so an unlimited number of users can operate simultaneously [60; 84]. A system user receives radio-frequency signals transmitted from known satellite locations. The time difference between when a signal is sent and when it is received is multiplied by the speed of light to determine a range difference. This measurement, termed "pseudorange," is *not* the true range.

The pseudorange differs from the true range primarily due to errors in the receiver clock. Since the receiver clock error is consistent for all measured pseudoranges over a reasonable time of interest, it is often considered a bias and must be estimated [84]. The three coordinates of spatial position must also be estimated. Normally, position determination requires the iteration of a system of equations based upon the four unknowns (three position coordinates and one clock bias term). Unless either system time or altitudes are accurately known, a minimum of four satellites in view is therefore required to determine position [60].

The actual GPS signal is not described here, as the importance in this research is in what the signal represents in terms of aiding measurements. For specific details on the GPS signal, see [49; 60; 84; 107]. GPS receivers can provide up to four different *raw* measurements: Pseudorange, Doppler, Carrier-phase, and Carrier-to-noise density. Raw measurements are generated internally, and are not to be confused with the position and velocity outputs provided by virtually all receivers. Access to raw measurements is required in many of the GPS/INS methods discussed in Section 2.6. The pseudorange measurement is used in nearly all applications, and is commonly the only measurement used.

2.5.0.1 Pseudorange Measurements. As mentioned in Section 2.5, pseudorange is true range between the satellite and user plus the effect from a number of error sources. Previously, only the receiver clock bias was mentioned. In truth, possible range errors also include satellite clock errors, as well as errors due to atmospheric conditions and multipath. The pseudorange measurement can be expressed as [107]:

$$\rho = r + c(\delta t_u - \delta t_{sv}) + c\delta t_{\mathit{iono}} + c\delta t_{\mathit{tropo}} + c\delta t_{\mathit{mp}} + v \tag{2.69}$$

where

ρ = GPS pseudorange (meters)

r = True range from the user to satellite (meters)

c = Speed of light (meters/second)

δt_u = Receiver clock error (seconds)

δt_{sv} = Satellite clock error (seconds)

δt_{iono} = Errors due to ionospheric delay (seconds)

$\delta t_{\mathit{tropo}}$ = Errors due to tropospheric delay (seconds)

δt_{mp} = Errors due to pseudorange multipath (seconds)

v = Errors due to receiver noise (meters)

The ρ represents the line-of-site (LOS) distance, or range, between the receiver and the satellite, and can be alternatively expressed in a simplified model form [84]:

$$\rho = \|\mathbf{x}_u - \mathbf{x}_s\|_{true} + b + \varepsilon \tag{2.70}$$

where

ρ = GPS pseudorange (meters)

$\mathbf{x}_u$ = The receiver x,y,z coordinates in ECEF (meters)

$\mathbf{x}_s$ = The satellite x,y,z coordinates in ECEF (meters)

b = The receiver clock bias (meters)

ε = Error in the measurement

Atmospheric modeling and antenna design are used to mitigate the impact of these errors on ρ, leaving the presumably predominate error to be the clock bias. The error, can then be modeled as a clock bias term, b, and the uncorrected residual errors, ε, which are assumed to be more noise-like, on the pseudorange measurement. Since the pseudorange is a non-linear measurement, it is linearized about an initial approximated receiver position and bias, ρ_0, and solved iteratively, making use of the previously described methods. The formulation is given by [84], letting the true position, $\mathbf{x}_t$ and true bias term, b, equal the initial guess plus an error term, with $\mathbf{x}_t = \mathbf{x}_0 + \delta\mathbf{x}$ and $b = b_0 + \delta b$, respectively. The error terms form the correction terms to apply to the initial estimates. Letting ρ_c represent the corrected pseudorange, the linear equation is developed using a first order Taylor series approximation [24; 84; 89; 90]:

$$\begin{aligned} \delta\rho &= \rho_c - \rho_0 \\ &= \|\mathbf{x}_t - \mathbf{x}_0 - \delta\mathbf{x}\| - \|\mathbf{x}_t - \mathbf{x}_0\| + (b - b_0) + \varepsilon \qquad (2.71) \\ &\approx -\frac{(\mathbf{x}_t - \mathbf{x}_0)}{\|\mathbf{x}_t - \mathbf{x}_0\|}\delta\mathbf{x} + \delta b + \varepsilon \end{aligned}$$

The equation can now be constructed in the form of $\mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{v}$, as shown, with $\mathbf{x} = [x\ y\ z\,]^T$ [84]:

$$\delta\boldsymbol{\rho} = \begin{bmatrix} \delta\rho_{m_1} \\ \delta\rho_{m_2} \\ \vdots \\ \delta\rho_{m_n} \end{bmatrix} = \begin{bmatrix} -\frac{(x_{m_1}-x_0)}{\|\mathbf{x}_{m_1}-\mathbf{x}_0\|} & -\frac{(y_{m_1}-y_0)}{\|\mathbf{x}_{m_1}-\mathbf{x}_0\|} & -\frac{(z_{m_1}-z_0)}{\|\mathbf{x}_{m_1}-\mathbf{x}_0\|} & 1 \\ -\frac{(x_{m_2}-x_0)}{\|\mathbf{x}_{m_2}-\mathbf{x}_0\|} & -\frac{(y_{m_2}-y_0)}{\|\mathbf{x}_{m_2}-\mathbf{x}_0\|} & -\frac{(z_{m_2}-z_0)}{\|\mathbf{x}_{m_2}-\mathbf{x}_0\|} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ -\frac{(x_{m_n}-x_0)}{\|\mathbf{x}_{m_n}-\mathbf{x}_0\|} & -\frac{(y_{m_n}-y_0)}{\|\mathbf{x}_{m_n}-\mathbf{x}_0\|} & -\frac{(z_{m_n}-z_0)}{\|\mathbf{x}_{m_n}-\mathbf{x}_0\|} & 1 \end{bmatrix} \begin{bmatrix} \delta\mathbf{x} \\ \delta b \end{bmatrix} + \boldsymbol{\varepsilon} \tag{2.72}$$

where the subscript m_i indicates the value is based on the i^{th} measurement. Representing $\delta\boldsymbol{\rho}$ as [84]:

$$\delta\boldsymbol{\rho} = \mathbf{H}\delta\mathbf{x} + \boldsymbol{\varepsilon} \tag{2.73}$$

where

$$\delta\mathbf{x} = \begin{bmatrix} \delta\mathbf{x} \\ \delta b \end{bmatrix}; \mathbf{H} = \begin{bmatrix} -\frac{(x_{m_1}-x_0)}{\|\mathbf{x}_{m_1}-\mathbf{x}_0\|} & -\frac{(y_{m_1}-y_0)}{\|\mathbf{x}_{m_1}-\mathbf{x}_0\|} & -\frac{(z_{m_1}-z_0)}{\|\mathbf{x}_{m_1}-\mathbf{x}_0\|} & 1 \\ -\frac{(x_{m_2}-x_0)}{\|\mathbf{x}_{m_2}-\mathbf{x}_0\|} & -\frac{(y_{m_2}-y_0)}{\|\mathbf{x}_{m_2}-\mathbf{x}_0\|} & -\frac{(z_{m_2}-z_0)}{\|\mathbf{x}_{m_2}-\mathbf{x}_0\|} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ -\frac{(x_{m_n}-x_0)}{\|\mathbf{x}_{m_n}-\mathbf{x}_0\|} & -\frac{(y_{m_n}-y_0)}{\|\mathbf{x}_{m_n}-\mathbf{x}_0\|} & -\frac{(z_{m_n}-z_0)}{\|\mathbf{x}_{m_n}-\mathbf{x}_0\|} & 1 \end{bmatrix} \tag{2.74}$$

facilitates use of the linear methods described earlier. For example, the least-squares solution for an overdetermined system can now be expressed as [84]:

$$\begin{bmatrix} \delta \mathbf{x} \\ \delta b \end{bmatrix} = \left(\mathbf{H}^T \mathbf{H}\right)^{-1} \mathbf{H}^T \delta \boldsymbol{\rho} \tag{2.75}$$

The detail provided in this section was partly designed to provide necessary familiarity with the GPS pseudorange measurement, as well as demonstrate some of the methodologies already presented for later discussions of GPS integrity algorithms. Another reason was to explicitly show the construction of the range measurements (e.g. $\delta\rho$) and the $\mathbf{H}$ matrix for comparison with image-aided measurement models. The next section provides a basic overview of two predominant INS/GPS integration schemes for visualization of INS aiding methodology prior to introducing concepts related to image aiding.

2.6 INS Integration

As described in the last section, the complementary properties of INS and GPS make integration schemes an attractive way to overcome the shortcomings of the individual navigation systems. This section provides a brief overview of ways an INS (or more specifically, the IMU) and GPS receiver may be combined to enhance the overall navigation solution.

Kalman filtering is the primary method of integrating INS and GPS data in most air navigation applications [14; 73; 80]. Either a direct or indirect Kalman filter formulation may be chosen based on the application. In the direct filter formulation, the filter estimates the total states of interest using the INS and GPS outputs as measurements. These states typically include, at a minimum, the position, velocity, and attitude of the vehicle. The filter uses a full dynamics model of the systems, and must have knowledge of how each subsystem is related functionally to the other. Consequently, a high sample rate is necessary in order to achieve adequate performance. In general, this implementation is computationally burdensome, and the filter performance may be compromised due to nonlinearities in the best models for vehicle dynamics. The direct formulation also presents a risk in that if the filter fails, the entire navigation system fails [80].

To limit the risks associated with the direct formulation, typical air navigation systems employ the indirect (error-state) approach to Kalman filtering [14; 73; 80]. The indirect formulation estimates the *errors* in the INS position, velocity, and attitude states, rather than the states themselves. Primarily dominated by low-frequency Schuler rate dynamics, INS errors change at a lower frequency than vehicle position, velocity, and attitude, allowing slower sample rates and reduced computational burden [80].

In addition to the two types of formulation, there are two possible mechanizations: feedforward and feedback. Given the previous discussion, these mechanizations will only be related to the indirect formulation. In the feedforward approach, the Kalman filter produces optimal estimates of the error states, which are subtracted from the INS-generated position to form a total state estimate. Since the INS is not dependent on the filter directly, position solutions would still be available, should the filter or sensors fail [80]. The feedforward approach does have one glaring weakness. Since corrections are not being applied to the INS directly, the INS errors will eventually grow large enough to violate the linear model assumptions made in using the Kalman filter. In the feedback implementation, the optimal estimates of the error state are applied to the INS as corrections used to bound the inertial errors. In this way, the adequacy of the linear model assumption is enhanced [80]. This corrective feedback becomes increasingly necessary when employing lower quality INS. The feedback implementation does run the risk of corrupting the INS output by feeding back inaccurate error estimates, but it is assumed that slow INS error dynamics would allow detection and compensation of such difficulties before the solution is degraded [80].

In the last several years, two particular INS/GPS integration methods have been consistently used to improve the overall accuracy of the navigation solution. Referred to as “loosely-coupled” and “tightly-coupled” integration, these techniques have gained wide acceptance [73], and are currently employed in many Air Force platforms. These methods are described in the sections that follow.

2.6.1 ***Loosely-Coupled Integration.*** Loosely-coupled integration is the the most straightforward implementation of the three integration schemes, since the GPS and INS typically already exist as independent navigation systems. The GPS receiver uses its own Kalman filter to process raw pseudorange (ρ) and delta-range (ΔR) measurements and to output a GPS navigation estimate of position and velocity. A second Kalman filter uses the navigation estimate as a "measurement" vector to produce estimates of the INS error. This second filter is an integration filter designed using both INS and GPS dynamics models. The resulting error estimates can be used in either a feedforward or feedback configuration. Figure 2.4 shows an example of the indirect-feedforward implementation. A feedback configuration is also possible, but not shown. Note that integration here is performed at the navigation solution level of each separate device, the INS and the GPS.

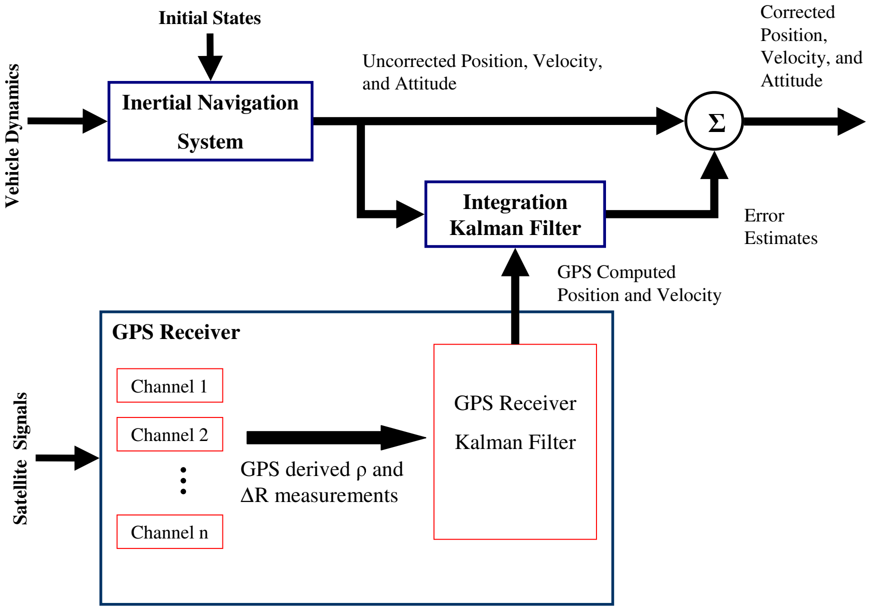
Initial States
Vehicle Dynamics
Inertial Navigation System
Uncorrected Position, Velocity, and Attitude
Corrected Position, Velocity, and Attitude
Σ
Integration Kalman Filter
Error Estimates
GPS Computed Position and Velocity
GPS Receiver
Satellite Signals
Channel 1
Channel 2
Channel n
GPS derived ρ and ΔR measurements
GPS Receiver Kalman Filter

Figure 2.4: Loosely Coupled GPS/INS Integration: Indirect-Feedforward Implementation.

Although this integration method typically yields better navigation solutions than either a stand-alone GPS receiver or INS, it is less accurate than the other two integration schemes. The degraded accuracy stems from the cascaded filter configuration. In a Kalman filter implementation, the measurement noise is assumed to be uncorrelated, Gaussian white noise. Since the measurements are first processed by the GPS filter, the "measurement noise" entering the second filter is time-correlated and must be modeled in that filter. This motivates the next discussion, tightly coupled integration.

2.6.2 ***Tightly-Coupled Integration***. The tightly-coupled configuration typically uses only a single centralized Kalman filter [73]. Removal of the intermediate GPS receiver Kalman filter reduces measurement noise correlation and enhances performance. The centralized Kalman filter combines an INS-computed solution with raw GPS measurements of pseudorange and possibly delta-pseudorange, provided by the receiver tracking loops. Here the integration is being performed at the pseudorange (and possibly delta-pseudorange) level. One advantage of the tightly-coupled configuration is that integrated system solutions may be possible when fewer than four satellites are available [85], unlike the case of GPS-alone operation or loosely-coupled systems. Increased resistance to GPS signal interference and improved performance in high vehicle dynamics environments are other advantages. The tightly-coupled system design and implementation are more complicated than a loosely-coupled system, but the payoffs in performance are significant. Although a feedforward configuration is also possible, only the feedback method is shown in Figure 2.5.

2.7 ***Image-Aiding***

The focus of this research is on investigating an integrity approach in image-aided INS. With this in mind, the emphasis is on the image sensor measurements (in the form of target pixel coordinates) that are provided to a Kalman filter used to integrate the imaging camera and the INS, and on the impact of faulty measurements on the integrated system.

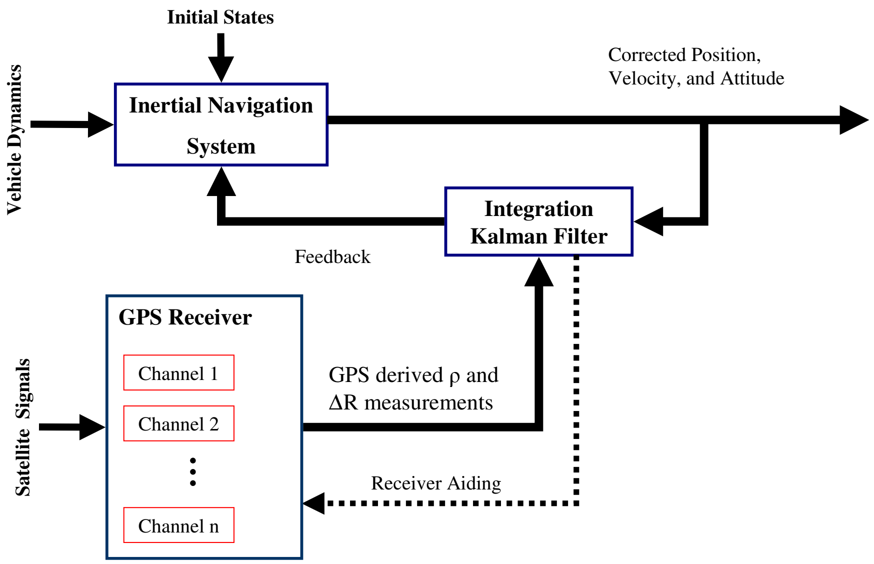
Initial States
Vehicle Dynamics
Inertial Navigation System
Corrected Position, Velocity, and Attitude
Integration Kalman Filter
Feedback
GPS Receiver
Channel 1
Channel 2
Channel n
Satellite Signals
GPS derived ρ and ΔR measurements
Receiver Aiding

Figure 2.5: Tightly-Coupled GPS/INS Integration: Indirect-Feedback Implementation.

This section provides some of the fundamental background on the correlation between the real world target and its representation as a sensor measurement. Since the primary concern is on the measurement provided by the imaging system, and not the imaging process itself, the following assumptions are made:

1. The feature tracker provides measurements in the form of a pair of pixel coordinates for a tracked target with the measurements provided at a suitable frame rate
2. The camera is calibrated allowing a precise relationship between pixel coordinates and the corresponding position in the camera frame, with necessary corrections for lens distortion
3. The association between a tracked target and given pixel coordinate is also known
4. The relationship between the camera mounting and vehicle body frame axes is known *a priori* and can be expressed as a direction cosine matrix, C_c^b

2.8 *Image-based Navigation Concepts*

The strength of the integration of INS with GPS, based on their complementary nature, is well-recognized. However, there are situations where GPS signals may not be available (e.g., signal obstruction or interference) [99]. As a result, an exploration of alternate navigation systems has brought recent research to focus on the potential of images provided by platform mounted imaging systems to help navigate. Image-based navigation can in fact be done without the use of an INS system, but the synergy between a tightly-coupled INS/image-aided system helps to bound anticipated INS error growth and provides the potential for a degree of accuracy comparable to that of civilian GPS positioning.

Image-based navigation may use active (e.g., laser scanners or radar) and/or passive (e.g., cameras) sensors to image an environment and then process series of image data to navigate, possibly even in unknown surroundings [48]. The main purpose of image-based navigation is to determine the position and orientation of the imaging sensor, which is assumed to have a known position and orientation relative to the platform that is carrying it [48]. In this research, the emphasis is on the passive camera sensors, since a goal of military operations is to operate with minimum detection. Additionally, there are currently many good cameras in use on operational systems. In general terms, a camera takes an image of a real three-dimensional (3-D) object and projects it onto a two-dimensional (2-D) image plane.

The complication with this projection is that motion cannot be determined from a single image [48], and mapping of a 3-D object onto the image plane is a many-to-one transformation [9]. So, a 3-D point of interest can not be recovered from its 2-D image unless something more is known about the object point (e.g., the range between the observer and sensor) or if there are images taken from two images in a single epoch that are a fixed distance apart [9; 119]. To overcome this limitation, image-based navigation is accomplished using primarily two fundamental approaches; either processing sequences of images from a single camera, requiring estimation of the movement/orientation between the images, or using multiple cameras in a known relative configuration designed to imitate human vision with parallax aiding depth determination. In both cases, there is higher uncertainty in the parameter reflecting the distance to the target, than in the relative bearing to the targets. Without the use of additional mapping information (e.g., predefined target coordinates, active sensor aiding, or surface elevation map), the measurements provided by the camera system can be considered primarily bearing-only measurements in the single camera case being considered in this work. These are key considerations in the image-based tracking methods described later in the section, but first an overview of perspective projection theory and camera-frame modeling are presented to aid further discussions.

*2.9 **Image Projection Theory***

The development of the digital camera has proven very valuable in providing nearly instantaneous image representations of a 3-D world scene. Variations of light are captured through the camera lens and focused on an image plane to form a 2-D representation of the scene. Real camera lenses have varying degrees of curvature, which allow a different fields of view of the scene being observed. However, pronounced curvature creates radial distortion, negatively affecting the image representation on the image plane. It is often useful to model of the camera lens as a "thin lens," or one that causes no radial distortion, such that light rays passing through the lens travel in straight lines, as shown in Figure 2.6.

Numerous calibration and correction techniques exist to mitigate radial distortion (see [42; 63; 77; 88; 119; 132]), facilitating modeling using the thin lens for a simplified interpretation of what is occurring in the image projection. The details are not presented here, but, fundamentally, the calibration involves using lens parameters, including the radius of curvature in all directions, to transform the warped or otherwise distorted image to a planar representation by applying a coordinate correction term to each pixel location. The following discussions assume that corrections have been applied removing image

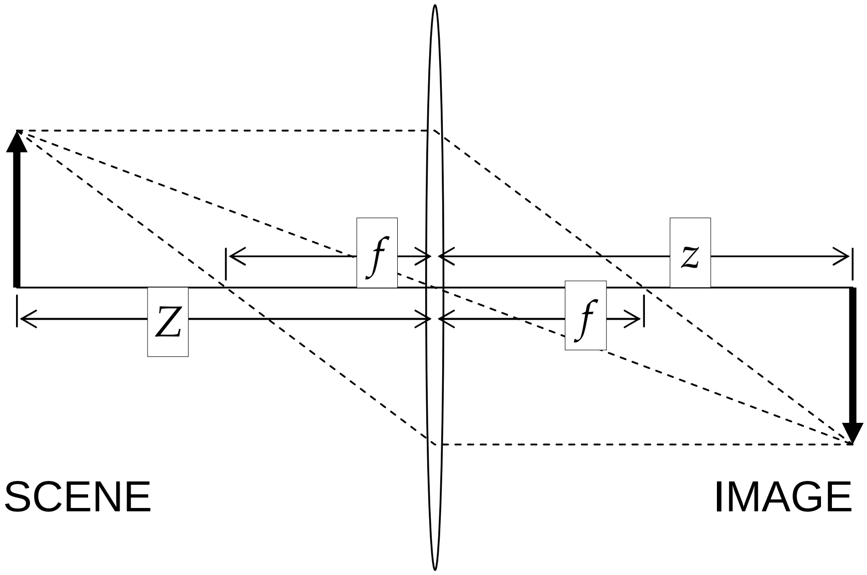

Figure 2.6: Thin lens camera model directing parallel light rays toward the focus, resulting in an image (Figure not to scale) [119].

effects due to the lens. The equation for the thin lens model, based on the parameters identified in Figure 2.6, is given by the following equation [77]:

$$\frac{1}{Z} + \frac{1}{z} = \frac{1}{f} \tag{2.76}$$

where f is the fixed camera focal length, and a known intrinsic camera parameter, while Z and z are the distances to the object of interest and its associated image, respectively.

To further aid in geometry and mathematical modeling, the camera lens aperture is reduced to a single point, such that all rays passing through the lens converge at this point, which is called the *perspective center* [51]. This effect of this reduction is that the image is projected one focal length behind the lens, such that $z = f$. Actual devices can be produced that approximate this reduced aperture model, which is often called the pinhole camera model. However, such devices suffer from diffraction effects and have very limited energy penetrating the lens (for projection), thus the pinhole camera is simply a good approximation of a well-focused imaging system [77].

In this model, the perspective center then becomes useful as the origin for a 3-D right-hand, orthogonal reference frame for the camera space, denoted as the camera-frame, or *c-frame* . As previously shown Figure 2.6, the image is actually projected behind the lens, and inverted with respect to the scene. For convenience, a transformation is typically done to work with a "positive image plane," which is accomplished by multiplying the original pixel coordinates by -1 and translating the image focal point from $-f$ to $+f$ ($+$ being in the direction of the scene) on the optical axis [77]. The resulting representation of the *c-frame* and the positive image plane are shown in Figure 2.7. The *c-frame* coordinate axes identified by a subscript c and and optical center as the origin, represented by O.

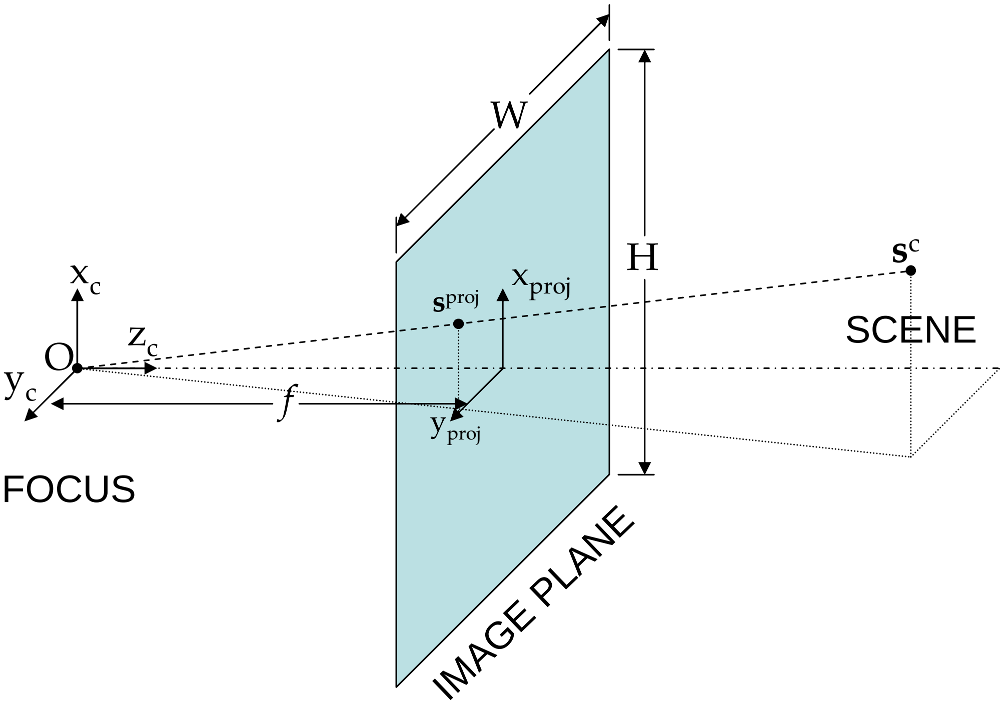

Figure 2.7: Camera projection model. The modified pinhole camera model places a virtual image plane one focal length in front of the perspective (or optical) center, eliminating the image inversion that occurs in the standard pinhole camera model [119].

Now the relationships between the point source $\mathbf{s}^c = [s^c_x, s^c_y, s^c_z]^T$ (Figure 2.7), and the perspective projection on the image plane, $\mathbf{s}^{proj} = [x^{proj}, y^{proj}, f]^T$ can be developed, using the notation consistent with [119]. As illustrated in Figure 2.7, there is a geometric relationship of similar triangles when the rays originate from the camera frame origin. By the properties of similar triangles, the corresponding angles of the two triangles formed by these coordinates are the same, and the lines forming the sides of the two triangles are proportional to each other. The second property is essential to the perspective projection theory, leading to the camera-centered coordinate system (i.e., the z^c axis is aligned with the optical axis and the perspective center is the focal point) projection equations:

$$x^{proj} = \frac{f}{s^c_z} s^c_x \;\; ; \;\; y^{proj} = \frac{f}{s^c_z} s^c_y \;\; ; \;\; z^{proj} = \frac{f}{s^c_z} s^c_z = f \tag{2.77}$$

where $\dfrac{f}{s^c_z}$ represents a scale factor, using the proportion of the triangle sides along the least observable parameter of the projection, with s^c_z being the distance between the point source and the perspective center along the z^c axis [119]. Letting $\mu = \dfrac{f}{s^c_z}$, the relationship can be rewritten in vector form as:

$$\mathbf{s}^{proj} = \mu \mathbf{s}^c \tag{2.78}$$

The well-focused image is captured via an imaging array aligned with the image plane [88]. In simple terms, the digital image recording process transforms the light rays received through the lens into an array of discrete values that can be measured and stored in a computer accessible format, referenced by pixel coordinates (x^{pix}, y^{pix}). The $M \times N$ pixel imaging array, with physical dimensions $H \times W$, has a planar coordinate system, with the origin defined to be the upper left corner, such that the center of uppermost, leftmost, pixel has the coordinate $(1, 1)$, with pixel values increasing down and to the right [119]. Figure 2.8 illustrates this relationship.

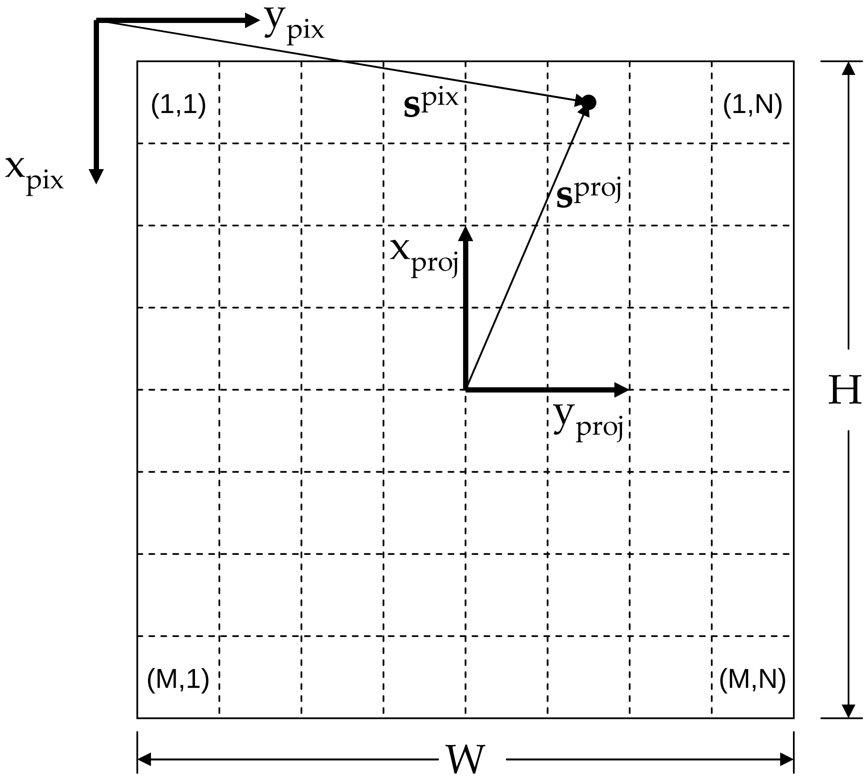

Figure 2.8: Illustration of camera image array of $M \times N$ pixels, with physical dimensions $H \times W$. Not to scale; image plane positions on camera exaggerated to emphasize relationship [119].

Since a projection of the camera frame onto the image plane assumes that the origin would be centered in the plane, at the optical center (in the calibrated case), a transformation must be made between the pixel coordinates and the projection coordinates. The projected coordinates are scaled by the pixel size and translated to the pixel coordinate frame axes, as described by [119]:

$$\mathbf{s}^{pix} = \begin{bmatrix} -\frac{M}{H} & 0 & 0 \\ 0 & \frac{N}{W} & 0 \end{bmatrix} \mathbf{s}^{proj} + \begin{bmatrix} \frac{M+1}{2} \\ \frac{N+1}{2} \end{bmatrix} \tag{2.79}$$

Combining Equations (2.78) and (2.79) produces a linear transformation from the camera frame to the pixel frame:

$$\underline{\mathbf{s}}^{pix} = \mu \begin{bmatrix} -\frac{M}{H} & 0 & \frac{M+1}{2f} \\ 0 & \frac{N}{W} & \frac{N+1}{2f} \\ 0 & 0 & \frac{1}{f} \end{bmatrix} \mathbf{s}^{c} \tag{2.80}$$

$$= \mu \mathbf{T}_c^{pix} \mathbf{s}^c \tag{2.81}$$

where $\mathbf{T}_c^{pix}$ is the homogeneous transformation matrix from camera frame to pixel frame [119]. This matrix defines the *interior orientation*, or relationship between the camera frame and image plane [48], since all of the parameters in $\mathbf{T}_c^{pix}$ are functions of the particular camera used. If the focal length is known and fixed, it can be normalized to 1, resulting in a system corresponding to an ideal pinhole camera, with the image plane positioned in front of the perspective center, at $f = 1$. The scale factor μ then simply becomes $\frac{1}{s_z^c}$.

The $\underline{\mathbf{s}}^{pix}$ vector is the homogeneous pixel coordinate vector that represents the measurements provided by the camera system. The matrix $\mathbf{T}_c^{pix}$ is square and full-rank, and is thus invertible, with the inverse represented as $\left(\mathbf{T}_c^{pix}\right)^{-1} = \mathbf{T}_{pix}^c$. As a result, if the pixel coordinates, $\underline{\mathbf{s}}^{pix}$, are measured, the camera frame coordinate can be determined by [88]:

$$\mathbf{s}^c = \frac{1}{\mu}\mathbf{T}_{pix}^c\underline{\mathbf{s}}^{pix} \tag{2.82}$$

or for the vector s^c normalized by its z-component and $f = 1$ [122]:

$$\underline{\mathbf{s}}^c = \mathbf{T}_{pix}^c\underline{\mathbf{s}}^{pix} \tag{2.83}$$

The optical measurements are essentially limited to bearings-only measurements due to the loss of scale when the range along the z-component axis is unknown, which is the case in single camera (monocular) imaging without aiding of an active ranging sensor or another means to determine depth of view, such as depth from focus or use of known object sizes. However, any observables can aid navigation, as they provide a line-of-sight representing a geometric locus of all points consistent with a given measurement [48].

Having described the relationship of the image projection in the camera frame (*interior orientation*), the full geometry of the relative position vectors (*exterior orientation*) in a navigation frame can be visualized. Table 2.1 describes the parameters of interest in the navigation frame, with the associated vectors shown in Figure 2.9. The geometric relationship is then described by:

$$\begin{aligned}
\mathbf{t}_m^n &= \mathbf{s}_m^n + \mathbf{d}^n + \mathbf{p}^n \\
\mathbf{s}_{\mathbf{pt}_m}^n &\triangleq \mathbf{t}_m^n - \mathbf{p}^n \quad = \quad \mathbf{s}_m^n + \mathbf{d}^n \\
\mathbf{s}_{\mathbf{pt}_m}^n &= \mathbf{C}_b^n\mathbf{C}_c^b\mathbf{s}_m^c + \mathbf{C}_b^n\mathbf{d}^b \quad = \quad \mathbf{C}_b^n\left(\mathbf{C}_c^b\mathbf{s}_m^c + \mathbf{d}^b\right)
\end{aligned} \tag{2.84}$$

Table 2.1: Parameter Definitions

Parameter	Description
$\mathbf{p}^n$	Vehicle position in navigation frame (NED)
$\mathbf{v}^n$	Vehicle velocity in navigation frame (NED)
$\mathbf{C}_b^n$	Vehicle body-to-navigation frame DCM
$\mathbf{s}_m^n$	Line-of-sight vector from the camera to the m^{th} target
	(vector for each landmark currently tracked)
$\mathbf{t}_m^n$	Location of the m^{th} target (vector for each landmark currently tracked)
$\mathbf{d}^b$	IMU-to-Camera leverarm offset
$\mathbf{C}_c^b$	Camera-to-body frame DCM

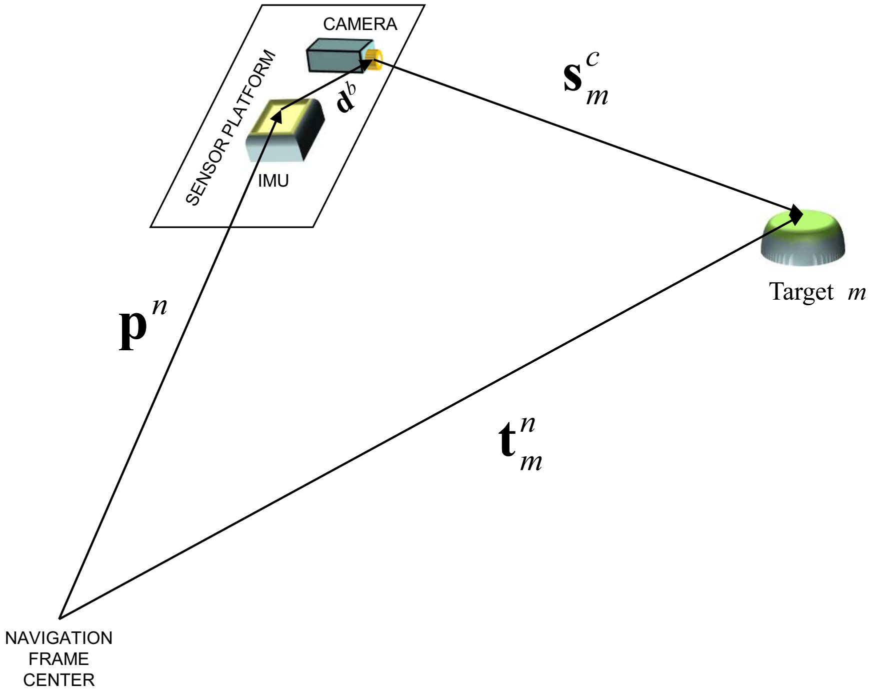

Figure 2.9: Vector geometry in the navigation reference frame (based on [119].)

Recalling that direction cosine matrices, $\mathbf{C}_{\beta}^{\alpha}$, are unitary, with the inverse equal to the transpose $\Big($ i.e. , $\left(\mathbf{C}_{\beta}^{\alpha}\right)^{-1} = \left(\mathbf{C}_{\beta}^{\alpha}\right)^{T} = \mathbf{C}_{\alpha}^{\beta}\Big)$, Equation (2.84) can be rewritten in terms of $\mathbf{s}_{m}^{c}$, matching the equation given by [122]:

$$\mathbf{s}_{m}^{c} = \mathbf{C}_{b}^{c}\mathbf{C}_{n}^{b}\left(\mathbf{t}_{m}^{n} - \mathbf{p}^{n}\right) - \mathbf{C}_{b}^{c}\mathbf{d}^{b} \tag{2.85}$$

or using the shorthand notation introduced in Equation (2.84):

$$\mathbf{s}_{m}^{c} = \mathbf{C}_{b}^{c}\mathbf{C}_{n}^{b}\left(\mathbf{s}_{\mathbf{pt}_{m}}^{n}\right) - \mathbf{C}_{b}^{c}\mathbf{d}^{b} \tag{2.86}$$

This section developed some necessary background in projection theory, and how pixel coordinates represent a target sighted by the camera. Combined with ability to transform between different reference frames to mathematically describe relative positions of the vehicle, sensor (camera), and target, the potential of image-aiding is clear. The interior orientation is a function of the camera sensor systems and calibration techniques, and is considered to be *a priori* knowledge. The exterior orientation is motion dependent, and a function three translations in position and three rotations in attitude, values which are better constrained in the estimate of motion by the integration of an INS. The question of how certain pixel locations are selected as a potential representation of desired targets is yet to be discussed. To the human eye, points of interest in a captured image would be easy to identify and associate between multiple images. However, with no human in the tracking loop, as is desired for most navigation system, it is necessary to automatically (via image/data processing) identify potential targets from the image and determine target consistency between images. Image feature detection and tracking methods are available to perform these tasks.

2.10 Feature Tracking

The preceding material described the projection of real-world 3-D objects of interest (also referred to as targets) into a 2-D image representation of the observed scene. Locations of the 3-D objects in the users real-world frame is often called a *map*, and actual map coordinates may or may not be known *a priori*. As mentioned, a single image does not provide information regarding motion of objects in the view, or, from an alternate point of view, a single image does not provide information about camera movement against stationary targets; the latter case is the perspective of image-based navigation. Therefore, multiple images from different vantage points, i.e., image sequences from a single camera or synchronous images from multiple cameras, must be used to estimate this motion. The use of multiple images requires "feature correspondence," or matching representations of observed objects from one image to another. So, an image-based navigation system must have the ability to identify objects of interest and correlate them between images in order to produce sufficient information to infer the motion that occurred.

This process is often called feature tracking in summary, but consists of a number of steps including feature detection, feature extraction, and feature tracking stages. Feature tracking is a very broad and deep research area unto itself, as it enables great number of imaging applications, including basic image registration, augmented reality, computer graphics interfacing, and medical system imaging, to name a few. Feature tracking is also critical to the INS/image-aiding navigation system that supports this research, but is assumed to take place prior to the point of interest in this study, which is the integrity algorithm based on the measurements provided to the integration filter. Many feature tracking approaches have been proposed in literature, dating as far back as 1959 [41], with advantages and disadvantages to each based on specific applications and assumptions. Therefore, the following discussion is not all encompassing, but presents a general characterization of selected feature tracking concepts that have been successfully applied to image-aided navigation; for additional details on feature tracking, the reader is referred to [22; 41; 43; 48; 63; 75; 76; 77; 83; 97; 113; 119; 133].

Images formed by the imaging array are representations of the observed scene, developed as a function of light passing through the lens and accumulating at each pixel location in the array. Each 2-D pixel coordinate (x, y) in the array has a value corresponding to the quantity of light accumulated, or technically, the quantization of the number photons arriving, at the time of the image capture. This intensity value is represented by $i(x, y)$, with the total image denoted as I_k, where k is the index of the image in a sequence. A very diverse scene would be expected to have a large range of intensity values across the entire image, while a nondescript image, like a plain white wall, would have similar intensity values throughout the image, with local variations primarily attributed to noise or texture. This disparity in the potential captured scenes is a fundamental factor in the image correspondence problem and potential source of error in measurements provided to the navigation system.

Feature-based correspondence dissects the images into prominent regions, or *features*, that can be quantified through image processing techniques. Ideal characteristics for features include [48]:

Significance: The feature significantly differs from its surroundings

Interpretability: The feature supports interpretation of the image

Rarity: The feature is unique from others

Invariance: The feature is invariant under geometric and temporal changes

Stability: The feature has some immunity to image disturbances

Quality: The feature position can be determined without bias

The simplest features in an image would be the individual pixel values themselves. However, summarizing [113], the inherent difficulty in determining the displacement of a point between one image and another is that a single pixel cannot not be tracked, unless it is singularly distinct from all of its neighboring pixels, which is nearly impossible when considering image distortion and intensity changes due to camera motion and/or image noise [120]. As a result, characteristic groups of pixels are examined, instead of a more volatile single pixel. These groups, defined in subregions, $W(\bar{\mathbf{x}}, \bar{\mathbf{y}}) \subset I$, of the image, can then be used to identify features, considering the above criteria, where $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$ are vectors centered on (x, y) coordinates of interest.

For the image-aided navigation problem, using features strictly based on individual pixel intensities does not often provide the required significance and rarity, even when subregions are used. Therefore, additional image processing is done to create features that are more discernible and trackable image to image. Examining not only the pixel intensity, but also changes in pixel intensities in the image, leads to more salient features like edges or corners, which are simple descriptions of what the feature represents in the local region of interest. As will be shown, these features can then potentially be characterized as symbolic vector representations, or descriptors, of image regions, resulting in significant data reduction [48]. Use of these features can also reduce ambiguity and time required for correspondence processing compared to that of intensity-based features, as the number of potential mismatches between features is decreased due to increased significance. By tracking the relationship between the image frame center and selected features of the overall image, more information regarding the relative rotational and translational motion between the camera and the observed scene can be derived. Table 2.2 lists a few of the these feature detectors, and names commonly used to reference the methods in literature. There is a noticeable overlap in the detector types, illustrating the sometimes ambiguous differences between them.

Table 2.2: Examples of Feature Detectors

Feature Detector Type	Common Reference Names
Edge	Canny [21], Sobel (in [56]), Prewitt (in [56]), Harris [46], Harris-Stephens (or Plessey) [46] Shi & Tomasi [105]
Corner	Harris-Stephens (or Plessey) [46], Features from Accelerated Segment Test (FAST) [101], Shi & Tomasi [105], Laplacian of Gaussian (LoG) (in [41]), Difference of Gaussian (DoG) (in [76])

Edge detection is of fundamental importance in image analysis due to its ability to characterize object boundaries and segment the scene [56]. An edge in an image is often readily discernible to the human eye viewing an image, as there is there is a recognizable structure created by a contrast between pixel intensities that separates two possibly very similar local regions in an image, forming a straight line. Figure 2.10 provides an illustration, where a photograph of a box has been taken in fairly close proximity to a similarly toned background. The figure also includes a close up of a selected edge line and a three-dimensional plot of the grayscale values at the selected pixel locations showing the changes in intensity.

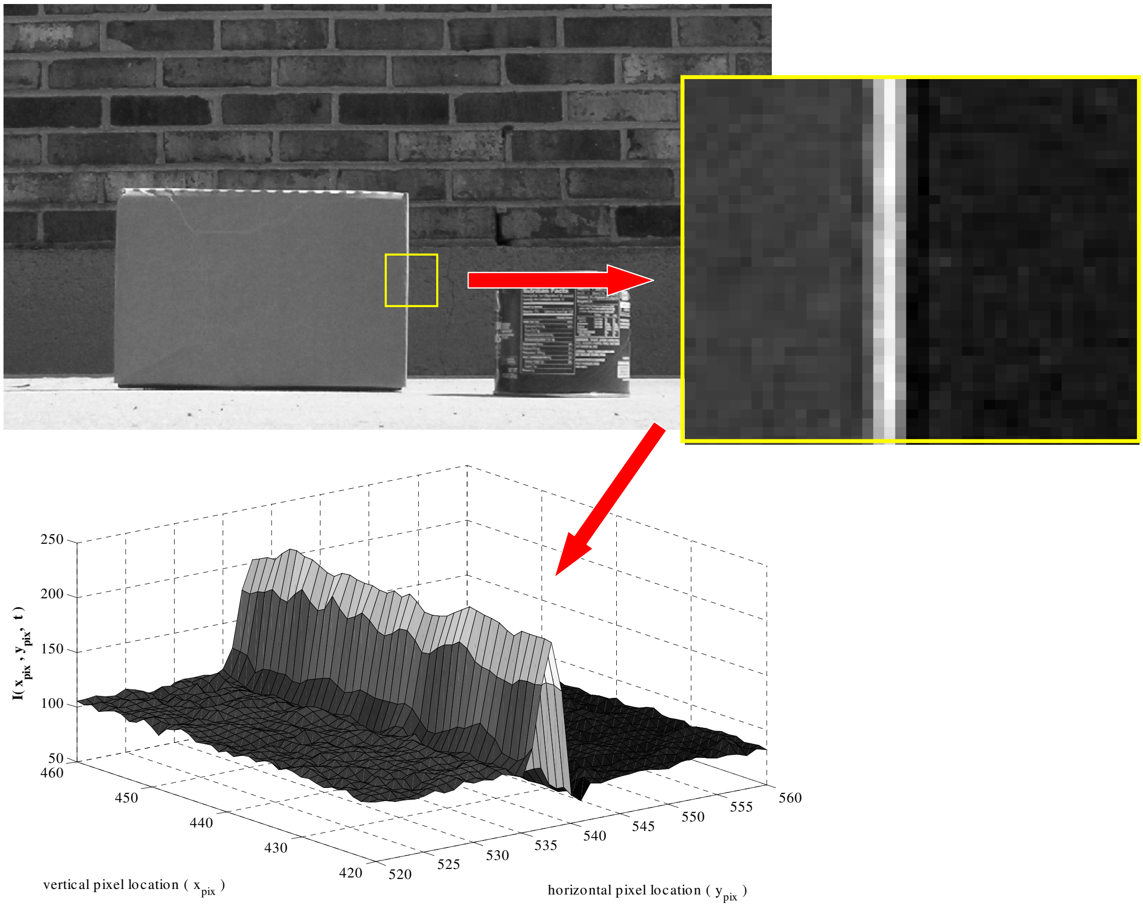

Figure 2.10: Example of Edge of Object in View.

This simple demonstration illustrates how information stored in an image is available for translation into metrics by which features may be identified, in numerical terms. It is worth noting that the magnitude of the edge in the lower left plot is enhanced by the fact that ambient light strikes the box on the righthand side, giving greater contrast to the grayscale values along this edge. Under different lighting conditions, transition in the magnitude of the grayscale levels would still be observable, but not as distinct. This effect was intentionally included to show how perspective changes may influence feature tracking algorithms, for better or for worse, depending on the conditions encountered.

In actual applications, feature detectors typically involve gradient or Laplacian operators to detect abrupt changes in the values at the different pixel coordinates. In either case, the image is generally smoothed using a Gaussian filter which weights pixel values at its center more strongly than at its boundaries. This reduces granularity caused by low level noise by smoothing the image [41]. The mathematics presented in this section are presented in continuous form for simplicity. In reality, the individual pixels are discrete samples of the observed scene, and are digitally processed using discrete sampling of the continuous signal. The Gaussian filter function is given by [76]:

$$g(x,y,\sigma) = \frac{1}{2\pi\sigma^2}\ e^{-(x^2+y^2)/2\sigma^2} \tag{2.87}$$

which is especially convenient for processing the 2-D image in each of the 1-D directions using the separability of the Gaussian (assuming isotropic, or $\sigma = \sigma_x = \sigma_y$):

$$g(x,y,\sigma) = g(x,\sigma)g(y,\sigma) = \frac{1}{\sqrt{2\pi}\sigma}\ e^{-x^2/2\sigma^2}\frac{1}{\sqrt{2\pi}\sigma}\ e^{-y^2/2\sigma^2} \tag{2.88}$$

The convolution of the image with the Gaussian kernel of a chosen standard deviation, σ, results in the filtered function $\ell(x,y,\sigma)$, defined by [76] as the scale space:

$$\ell(x, y, \sigma) = g(x, y, \sigma) * i(x, y) \tag{2.89}$$

Edge feature detection is accomplished by taking computing the image gradient vector $\nabla\ell(x, y) = [\ell_x, \ell_y]^T$, where:

$$\ell_x = \frac{\partial\ell(x, y)}{\partial x}, \quad \ell_y = \frac{\partial\ell(x, y)}{\partial y} \tag{2.90}$$

and evaluating $\eta = \|\nabla\ell(x, y)\|^2$, for $\sigma > 0$ [77]. If η exceeds a predetermined threshold τ, for $\tau > 0$, then the given pixel is on an edge and can be selected for tracking. Figure 2.11 shows the impact of the changing σ values on the presentation of the image gradient. It should be pointed out that differentiation is linear and shift invariant, and that smoothing with the Gaussian function and then differentiating is equivalent to convolving the image with the derivative of the smoothing function [41].

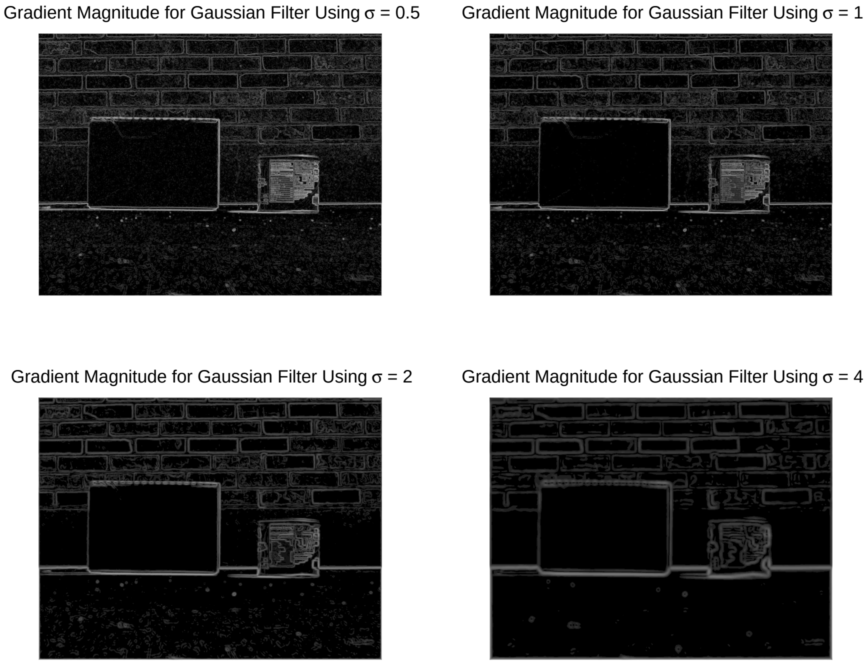
Gradient Magnitude for Gaussian Filter Using σ = 0.5
Gradient Magnitude for Gaussian Filter Using σ = 1
Gradient Magnitude for Gaussian Filter Using σ = 2
Gradient Magnitude for Gaussian Filter Using σ = 4

Figure 2.11: Illustration of Gradient Changes as Function of Gaussian Filter σ Value.

Corner feature detection extends the concept of edge detection by evaluating gradient behavior within the window, W. Corner feature detection is attributed to [46], and is represented in various forms. A characteristic matrix, $\mathbf{G}$, is formed by the products of the gradients in both the x and y directions [41]:

$$\mathbf{G} = \sum_{W} (\nabla \ell)(\nabla \ell)^T = \begin{bmatrix} \sum_W \ell_x^2 & \sum_W \ell_x \ell_y \\ \sum_W \ell_y \ell_x & \sum_W \ell_y^2 \end{bmatrix} \tag{2.91}$$

Conceptually, in a window with little variation in image intensity, the eigenvalues of $\mathbf{G}$ will be small, reflecting the lack of discernible gradients. However, in areas with reasonable contrast, both eigenvalues should be large, resulting from 2-D contributions of gradient changes from the pixels in the window. A corner feature is detected if the smallest eigenvalue of $\mathbf{G}$ [77; 105], or alternatively, if the ratio of the maximum eigenvalue to the minimum eigenvalue [26], is greater than some predetermined threshold, $\tau \in \mathbb{R}$. Other variations, including the well-known Harris corner detector [46], use the following quantity for evaluation against a threshold [46; 77; 119]:

$$C(\mathbf{G}) = det(\mathbf{G}) + k \left[trace^2(\mathbf{G}) \right] \tag{2.92}$$

where k is a tuning parameter used to provide discrimination against high contrast pixel step edges [133]. Small values of k favor detection of corner features, while increasing the value of k decreases the requirement of significant gradient variation in more than one direction [77].

One problem with all differential techniques is that they may fail if the perceived transition of intensity is bigger than a few pixels [77]. To counter this problem, a coarse-to-fine approach may be used, by varying the σ parameter and building a pyramid of smoothed and sub-sampled images. This "multi-scale" approach replicates changing the focus of the lens to bring out finer detail in the smoothed images, and is foundational to the scale-invariant feature tracking (SIFT) method proposed by [75; 76], which has become a successful and widely used approach to feature tracking [22; 37; 43; 47; 57; 61; 64; 75; 104; 109; 111; 115; 119]. SIFT is growing in popularity over the well-known Kanade-Lucas-Tomasi (KLT) [105; 113] tracking algorithm in applications where larger frame-to-frame differences between images are observed, as is likely the case of an aircraft mounted camera, unless an exceedingly high frame rate is used.

Figure 2.11 illustrated how the distinctiveness of the image gradient changed with variation in the σ parameter of the Gaussian smoothing function, and showed that a gradient feature may or may not be observable under different choices of σ. This concept is used in SIFT in employing a difference-of-Gaussian (DoG) technique to produce features and descriptors that are more robust to changes in illumination, image noise, rotation, scaling and viewpoint, all of which are critical considerations for image-aided navigation, especially for airborne platforms. There are four basic detection stages in SIFT [61; 76]:

1. Scale-space extrema detection (selection)
2. Keypoint localization
3. Orientation assignment
4. Generation of keypoint descriptors

In the first stage, key feature locations are detected using a DoG function, $D(x, y, \sigma)$, based on Equation (2.89), to construct image pyramids of different σ scale values and a constant multiplicative factor k [76]:

$$D(x, y, \sigma) = \ell(x, y, k\sigma) - \ell(x, y, \sigma) \tag{2.93}$$

with $k = 2^{1/s}$, $s \in \mathbb{Z}$, producing $s + 3$ images for each pyramid level. A common choice of values is $s = 2$ such that $k = \sqrt{2}$ [75; 76; 119]. A key feature search is conducted over all scales and pixel locations, in 3×3 grids, to determine the minima and maxima values of $D(x, y, \sigma)$, with a candidate keypoint selected only if the value is smaller or larger than the eight closest neighbors in the current image and the nine neighbors in both the images at the scale above and below the current scale.

The second stage involves testing the stability of these candidate keypoints. Keypoints exhibiting low contrast with the surrounding regions are rejected, as well as those failing to meet a threshold, η, criteria comparable to that described in Equation (2.92). Specifically, the following test is described [76]:

$$\frac{trace^2(\mathbf{G})}{det(\mathbf{G})} < \frac{(\eta + 1)^2}{\eta} \tag{2.94}$$

In stage three, image gradient magnitudes and orientations are sampled around the selected keypoint using a circular Gaussian weighting function based on the scale for the chosen keypoint. An orientation histogram is then constructed over eight gradient angles in $[-\pi, \pi]$, with entries weighted by the gradient magnitude and Gaussian weighting factor, which is multiplicative factor of the scale. This process functionally reduces the dimensionality of the region around the keypoint and improves rotational invariance. The final stage is the construction of feature vectors, z_j^*, which is done for each keypoint, j, by augmenting the 4×1 pose vector, $\mathbf{z}_j^{pose}$ with an $(m \times m \times n) \times 1$ normalized descriptor vector, $\mathbf{z}_j^{object}$, to form [119]:

$$\mathbf{z}_j^* = \begin{bmatrix} \mathbf{z}_j^{pose} \\ \dots \\ \mathbf{z}_j^{object} \end{bmatrix} \tag{2.95}$$

with the $\mathbf{z}_j^{pose}$ vector given by [119]:

$$\mathbf{z}_j^{pose} = \begin{bmatrix} x_j^{pose} \\ y_j^{pose} \\ \sigma_j \\ \theta_j \end{bmatrix} \tag{2.96}$$

where (x_j^{pose}, y_j^{pose}) are the pixel coordinates of the feature keypoint, σ_j is the scale, and θ_j is the primary orientation, all for the j_{th} keypoint. The term *pose* describes elements related to position and orientation of an object, while *object* refers to the object itself, which is ideally recognizable regardless of pose. The descriptor vector dimensions are determined from the histogram dimensions, where $m \times m$ is the histogram array size and n is the number of orientation bins in each array element. Figure 2.12 shows an example of the histogram relationship for the $2 \times 2 \times 8 = 32$ case. Analysis by [76], suggests that the $4 \times 4 \times 8 = 128$ structure commonly used performs best. This feature description is highly desirable as it contains separable *pose* and *object* information, as denoted by the superscripts in Equation (2.95), with complete independence being the ideal [119].

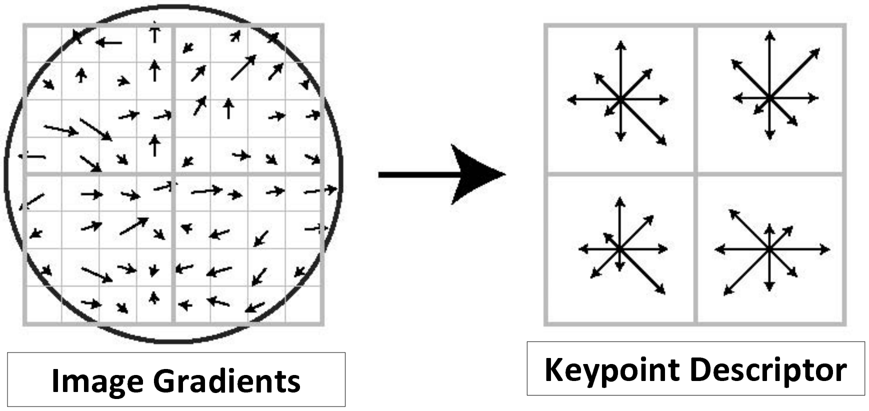

Figure 2.12: Illustration of Relationship between Image Gradient and Resultant Keypoint Descriptors [76].

The concept of SIFT has been extended in recent work by [61] and [83] to further invariance to affine image motion. In [61], the method applies principle component analysis (PCA) to the normalized image subregions instead of using the smoothed, weighted histograms of SIFT. PCA-SIFT uses the same keypoints as SIFT, and then extracts a large image patch (on the order of 41×41) in order to [61]:

1. Pre-compute an eigenspace to express the gradient images of image subregions
2. Compute the local gradients of the subregions
3. Project the gradient image vector using the eigenspace to derive a compact feature vector

The result is smaller descriptor vectors, simplifying and reducing computational burden in feature processing. The process described by [83] is known as gradient location-orientation histogram (GLOH) and is designed to improve robustness and distinctiveness of the SIFT descriptor. In this method, the SIFT descriptor is computed in log-polar coordinates on a grid with three radial lengths and eight angular envelopes. The gradient orientations are quantized into 16 bins, resulting in a 272 bin histogram, as compared to the 128 of SIFT. This descriptor is then reduced down to 128 using the largest 128 eigenvalues determined through PCA.

Once features are extracted, feature matching done to establish correspondence between images. Feature matching can be accomplished through a number of different ways, strongly based on pattern recognition and classification techniques. Two particular methods are consistently used in the approaches mentioned in this section, either a normalized cross-correlation or a Euclidian distance type measure. In general terms, the normalized cross-correlation (NCC) relationship is described by [64]:

$$\alpha(\mathbf{a}, \mathbf{b}) = \frac{\sum_i (a_i - \mu_{\mathbf{a}})(b_i - \mu_{\mathbf{b}})}{\sum_i (a_i - \mu_{\mathbf{a}})^2 \sum_i (b_i - \mu_{\mathbf{b}})^2} \tag{2.97}$$

where $\mathbf{a}$ and $\mathbf{b}$ are feature vectors made up of i elements and having a mean μ. Values of α above a predetermined threshold are accepted as matches. The Euclidean distance measure compares the similarity between the feature vectors, and is given by [82]:

$$D(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \mathbf{b}\| = \sqrt{(\mathbf{a} - \mathbf{b})^T(\mathbf{a} - \mathbf{b})} \tag{2.98}$$

When a covariance relationship is known, the covariance matrix, $\mathbf{P}$, can be used to weight the distance measures, as shown [82]:

$$D(\mathbf{a}, \mathbf{b}) = \sqrt{(\mathbf{a} - \mathbf{b})^T\mathbf{P}^{-1}(\mathbf{a} - \mathbf{b})} \tag{2.99}$$

A useful extension of the weighted form is called the Mahalanobis distance, and is given by [77]:

$$D(\mathbf{a}, \mu_{\mathbf{a}}) = \sqrt{(\mathbf{a} - \mu_{\mathbf{a}})^T\mathbf{P}^{-1}(\mathbf{a} - \mu_{\mathbf{a}})} \tag{2.100}$$

where similarity is evaluated based on the feature vector, $\mathbf{a}$, and its mean, $\mu_{\mathbf{a}}$. This association is used in inertial and image sensor fusion algorithms described in [119], for example, where $\mathbf{a}$ represents a feature vector taken at a measurement time and $\mu_{\mathbf{a}}$ is a Kalman filter prediction of the measurement.

The minimum discrepancy, distance, or cost between matched features from different images occurs when the features have exactly the same intensity and are aligned with exactly the same center and orientation. However, in real applications, this ideal is almost never met. Intensity variations have already been discussed. What must now be considered is that in the rigid-body motion presented in the image-aided navigation problem, observations made in an image, I_{k+j}, $k, j \in \mathbb{Z}$, may have undergone rotational and/or translational changes with respect to a previous image, I_k.

While SIFT and related feature tracking algorithms do not necessarily assume *a priori* information about the camera views, earlier feature matching techniques require a coplanarity assumption that is dependent on the camera views, and is made tractable based on the concept of epipolar geometry. Epipolar geometry defines a clear relationship between image observations made of the same real-world 3-D point seen in two different camera views. This relationship is shown in Figure 2.13, where a baseline connects the origin position of the camera frame, O_i, in two different instances, defining the epipoles, e_i, where this baseline intersects the respective images I_i. Remaining consistent with previous notation, the transformation between frames, $i = a \longmapsto b$, uses a rotation matrix denoted as $\mathbf{C}_a^b$, and a translation vector is denoted as $\mathbf{p}_{ba}^b$ with a condition that $||\mathbf{p}|| \neq \mathbf{0}$.

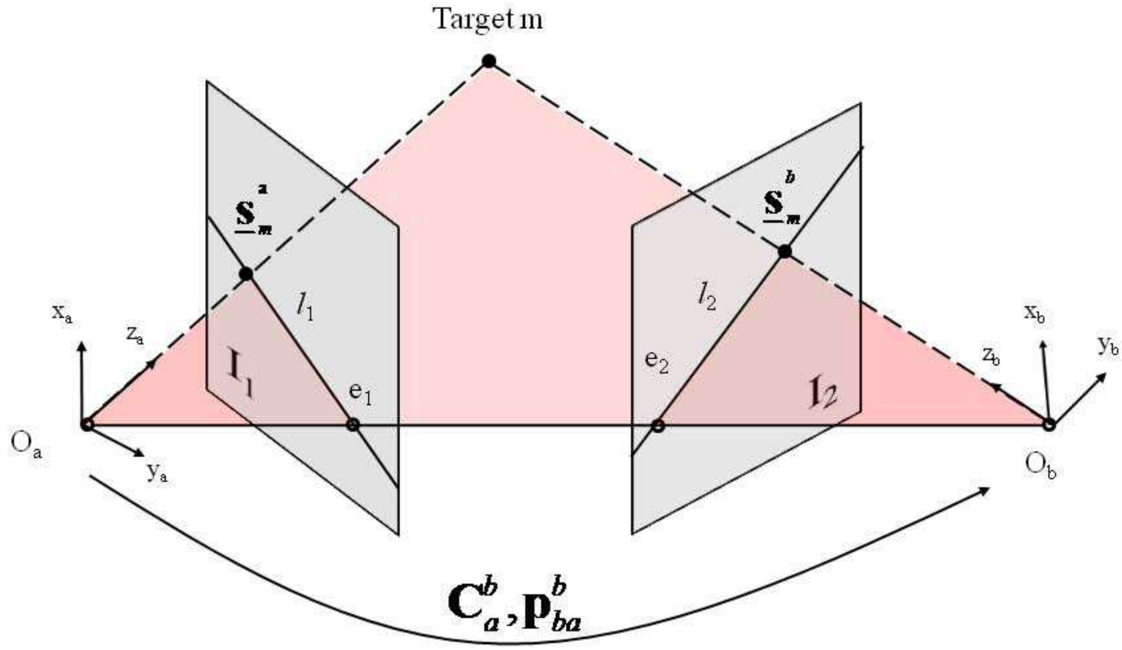
Target m
$\underline{\mathbf{s}}_m^a$
$\underline{\mathbf{s}}_m^b$
l_1
l_2
x_a
z_a
y_a
O_a
I_1
e_1
e_2
I_2
x_b
y_b
z_b
O_b
$\mathbf{C}_a^b, \mathbf{p}_{ba}^b$

Figure 2.13: Epipolar geometry between two images viewing same real-world point from changed camera positions.

The epipolar geometry establishes a constraint on the area where the target could actually be located and still be observed in both camera frames at the designated coordinates. The lines, l_i, are epipolar lines representing projections of the target rays (the line from each frame center to the target) from the opposite perspective, and provide a baseline region for where projections of the target should lie. For image correspondence, the equation of the epipolar constraint (assuming calibrated camera) is found in [74] and given using notation consistent with [123]:

$$(\underline{\mathbf{s}}_m^b)^T[\mathbf{p}_{ba}^b \times]\mathbf{C}_a^b(\underline{\mathbf{s}}_m^a) = 0 \tag{2.101}$$

where $[\mathbf{p}_{ba}^b \times]$ is a skew symmetric matrix. The quantity $[\mathbf{p}_{ba}^b \times]\mathbf{C}_a^b$ is often denoted as the 3×3 *essential matrix*, E. In image aiding processes that do not incorporate IMU information, it is possible to use recursive algorithms (e.g., random consensus sample [40] or an "eight point algorithm" [74]) to estimate motion from E. For example, the eight point algorithm process requires at least eight image correspondences, and uses singular value decomposition to solve for the rotation matrix and translation vector (for details see [77]). In terms of the epipolar geometry concept, a great advantage results from IMU integration, which can provide information about the rotation and translation between image times, precluding the necessity of solving the epipolar constraint through iteration. In a complimentary configuration, IMU/image aiding not only increase robustness and computational efficiency with the IMU helping provide the camera orientation, but also with the image system helping correct errors in the inertial system [128].

Before describing specifics of integrated techniques, it should be pointed out that feature matching errors can and will occur, regardless of the feature tracking technique used. Two primary sources of error are "bad locations" and false matches [133]. Bad locations are in fact incorrect pixel positions resulting from poor localization of the points. False matches occur when a features between images are incorrectly correlated. These incorrect features are commonly called outliers, and although the thresholding techniques described earlier can help reduce outliers, these errors are still quite possible. There is a fundamental trade-off in establishing a threshold; if it is set too low false matches may be passed, and if it is set too high, there not be enough confirmed matches to generate a solution. Incorporation of outliers into the estimation problem can corrupt the overall position estimation process, whether using an integrated system or not. In an integrated scheme, these errors are propagated through to the navigation filter and are motivating factors for the development of an image-aided integrity method.

2.11 INS/Image-Aiding Using Landmark States

Simultaneous Localization and Mapping (SLAM) is an image-based navigation process conceptualized about 20 years ago to solve the dual problem of building a consistent map of unknown environment while simultaneously determining vehicle position within the environment using probabilistic methods [34]. Initially, the vehicle was a robot, set to navigate in a nearly fixed plane and limited area environment. SLAM was accomplished by surveying and then revisiting observed landmarks in order to refine the estimated position of both the landmark and the position. Image-only SLAM has been well-studied, but continues to evolve. Over recent years, SLAM, or "SLAM-like" methods have been proposed for sensor fusion in air vehicle navigation in 3-D space and over broad areas, where revisitation of landmarks may or may not occur. In these cases, the emphasis is not on determining an absolute location, but rather relative positioning, which is particularly useful for aiding an INS.

INS/image-aided navigation, has been the subject of much recent research [5; 35; 51; 59; 86; 88; 115; 119]. This dissertation will employ methodology and notational convention described by Veth in [119; 121; 122; 123] and follow-on work by [88], assuming stationary landmarks. The approach by Veth represents the key attributes of IMU/image-aided algorithms described in other literature, and also employs stochastic feature prediction after SIFT feature extraction to reduce the feature correspondence search space, facilitate real-time implementation, and further decrease the possibility of false matches.

As with the GPS discussion earlier, the focus in the investigation of integrity is in the measurements, thus only the relevant details related to the measurement models are presented here. Although not explicitly shown by the equations that follow, estimation of the depth parameter for unknown landmarks cannot be made until at least two images have been recorded in the monocular vision case being studied. Also, additional sensors or digital elevation data, such as a baro altimeter or statistical terrain model, respectively, could augment the INS/image-aided system for better height estimation.

The formulation uses an extended Kalman filter with an error state vector based on errors present in the parameters described earlier in Table 2.1:

$$\delta\hat{\mathbf{x}} = \left[(\delta\hat{\mathbf{p}}^n)^T \;\; (\delta\hat{\mathbf{v}}^n)^T \;\; (\hat{\boldsymbol{\psi}})^T \;\; (\delta\hat{\mathbf{a}}^b)^T \;\; (\delta\hat{\mathbf{b}}^b)^T \;\; (\delta\hat{\mathbf{t}}_1^n)^T \;\; \ldots \;\; (\delta\hat{\mathbf{t}}_N^n)^T \right]^T \tag{2.102}$$

where N is the number of landmarks tracked. The perturbation error states are defined as [122]:

$$\begin{aligned}
\delta\mathbf{p}^n &= \tilde{\mathbf{p}}^n - \mathbf{p}^n \\
\delta\mathbf{v}^n &= \tilde{\mathbf{v}}^n - \mathbf{v}^n \\
\tilde{\mathbf{C}}_b^n &= (\mathbf{I} - [\boldsymbol{\psi}\times])\,\mathbf{C}_b^n \\
\delta\mathbf{b}^b &= \tilde{\mathbf{b}}^b - \mathbf{b}^b \\
\delta\mathbf{a}^b &= \tilde{\mathbf{a}}^b - \mathbf{a}^b \\
\delta\mathbf{t}_m^n &= \tilde{\mathbf{t}}_m^n - \mathbf{t}_m^n
\end{aligned} \tag{2.103}$$

with the tilde, $\tilde{(\cdot)}$, representing the nominal trajectory. The error state vector can be defined more compactly letting the first five error states be denoted as $\delta\hat{\mathbf{x}}_v$, a 15×1 vector for the vehicle error states of interest, and the final N error states denoted as $\delta\hat{\mathbf{x}}_t$, a $3N \times 1$ vector for the total number of landmark error states, such that:

$$\delta\hat{\mathbf{x}} = \left[\underbrace{(\delta\hat{\mathbf{p}}^n)^T \;\; (\delta\hat{\mathbf{v}}^n)^T \;\; (\hat{\boldsymbol{\psi}})^T \;\; (\delta\hat{\mathbf{a}}^b)^T \;\; (\delta\hat{\mathbf{b}}^b)^T}_{(\delta\hat{\mathbf{x}}_v)^T} \;\; \underbrace{(\delta\hat{\mathbf{t}}_1^n)^T \;\; \ldots \;\; (\delta\hat{\mathbf{t}}_N^n)^T}_{(\delta\hat{\mathbf{x}}_t)^T} \right]^T \tag{2.104}$$

$$\delta\hat{\mathbf{x}} = \begin{bmatrix} \delta\hat{\mathbf{x}}_v \\ \delta\hat{\mathbf{x}}_t \end{bmatrix} \tag{2.105}$$

with N actually varying over time as landmarks come into and leave the field of view. There must be a practical maximum on the number of landmarks tracked at any given time in order to maintain computational efficiency. Details of this process, called track maintenance, can be found in [119]. The measurement for the m^{th} feature is derived from the relationship first shown in Equation (2.83), with $\underline{\mathbf{s}}^{pix}$ replaced by $\mathbf{z}(t_i)$ [122]:

$$\mathbf{z}_m(t_i) = \mathbf{T}_c^{pix}\underline{\mathbf{s}}_m^c(t_i) + \mathbf{v}(t_i) \tag{2.106}$$

where $\underline{\mathbf{s}}_m^c(t_i)$ is the homogeneous form of Equation (4.2), and $\mathbf{v}(t_i)$ is independent additive white Gaussian noise of zero mean and covariance $\mathbf{R}$. The Jacobian of the nonlinear measurement function, $\mathbf{h}[\mathbf{x}(t_i)] = \mathbf{T}_c^{pix}\underline{\mathbf{s}}_m^c(t_i)$, is used to describe the linearized observation matrix [122]:

$$\mathbf{H} = \left[\frac{\partial\mathbf{h}}{\partial\mathbf{p}^n} \;\; \mathbf{0}_{3\times3} \;\; \frac{\partial\mathbf{h}}{\partial\boldsymbol{\psi}} \;\; \mathbf{0}_{3\times6} \;\; \frac{\partial\mathbf{h}}{\partial\mathbf{t}_1^n} \;\; \ldots \frac{\partial\mathbf{h}}{\partial\mathbf{t}_N^n} \right] \tag{2.107}$$

where the non-zero elements of $\mathbf{H}$ are the partial derivatives given by the following equations based on [119], with $\mu = \dfrac{1}{s_{m_z}^c}$ and $\boldsymbol{\beta} = [0 \;\; 0 \;\; 1]$:

$$\frac{\partial \mathbf{h}}{\partial \mathbf{p}^n} = \mu \mathbf{T}_c^{pix} \left(\underline{\mathbf{s}}_m^c \boldsymbol{\beta} \mathbf{C}_b^c \mathbf{C}_n^b - \mathbf{C}_b^c \mathbf{C}_n^b \right) \tag{2.108}$$

$$\frac{\partial \mathbf{h}}{\partial \boldsymbol{\psi}} = \mu \mathbf{T}_c^{pix} \left(\frac{\partial \mathbf{s}_m^c}{\partial \boldsymbol{\psi}} - \underline{\mathbf{s}}_m^c \boldsymbol{\beta} \frac{\partial \mathbf{s}_m^c}{\partial \boldsymbol{\psi}} \right) \tag{2.109}$$

where

$$\frac{\partial \mathbf{s}_m^c}{\partial \boldsymbol{\psi}} = -\mathbf{C}_b^c \mathbf{C}_n^b \left[\left(\mathbf{t}^n - \mathbf{p}^n \right) \times \right] \tag{2.110}$$

and finally

$$\frac{\partial \mathbf{h}}{\partial \mathbf{t}_m^n} = \mu \mathbf{T}_c^{pix} \left(\mathbf{C}_b^c \mathbf{C}_n^b - \left(\underline{\mathbf{s}}_m^c \boldsymbol{\beta} \mathbf{C}_b^c \mathbf{C}_n^b \right) \right) \tag{2.111}$$

For the integrity problem, the significance of this formulation is in the structure of the observation matrix compared to the GPS case. In the GPS case, information is known about the location of the satellite relative to the vehicle, which allows the measurement to be modeled as a pseudorange resulting in one $\mathbf{H}$ matrix row for each measurement. However, in the case of image aiding, the measurements are pixel pairs, resulting in multiple rows per measurement in the observation matrix (2-D on the pixel plane or 3-D homogenous in the camera frame). Another consideration is in the correlation between the INS and the vision system due to the mutual aiding. Using simplified notation based on the vehicle and target state relationship shown in Equation (2.104), let:

$$\mathbf{H}_{x_v} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}_v} \tag{2.112}$$

and

$$\mathbf{H}_{x_t} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}_t} \tag{2.113}$$

such that $\mathbf{H} = \begin{bmatrix} \mathbf{H}_{x_v} \vdots \mathbf{H}_{x_t} \end{bmatrix}$. The error covariance matrix is defined as:

$$\mathbf{P} = E\{\delta\hat{\mathbf{x}}\delta\hat{\mathbf{x}}^T\} = E\left\{\begin{bmatrix} \delta\hat{\mathbf{x}}_v \\ \delta\hat{\mathbf{x}}_t \end{bmatrix}\begin{bmatrix} \delta\hat{\mathbf{x}}_v \\ \delta\hat{\mathbf{x}}_t \end{bmatrix}^T\right\} \tag{2.114}$$

with the resultant covariance submatrices denoted:

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{\delta\hat{\mathbf{x}}_v\delta\hat{\mathbf{x}}_v} & \mathbf{P}_{\delta\hat{\mathbf{x}}_v\delta\hat{\mathbf{x}}_t} \\ \mathbf{P}_{\delta\hat{\mathbf{x}}_t\delta\hat{\mathbf{x}}_v} & \mathbf{P}_{\delta\hat{\mathbf{x}}_t\delta\hat{\mathbf{x}}_t} \end{bmatrix} \tag{2.115}$$

The pixel location uncertainty for m^{th} measurement (i.e. , $\delta\hat{\mathbf{x}}_t = \delta\hat{\mathbf{t}}^n_m$) the can then be described by:

$$\mathbf{P}_{\delta\mathbf{z}\delta\mathbf{z}} = \mathbf{H}_{x_v}\mathbf{P}_{\delta\hat{\mathbf{x}}_v\delta\hat{\mathbf{x}}_v}\mathbf{H}^T_{x_v} + \mathbf{H}_{x_v}\mathbf{P}_{\delta\hat{\mathbf{x}}_v\delta\hat{\mathbf{x}}_t}\mathbf{H}^T_{x_t} + \mathbf{H}_{x_t}\mathbf{P}_{\delta\hat{\mathbf{x}}_t\delta\hat{\mathbf{x}}_v}\mathbf{H}^T_{x_v} + \mathbf{H}_{x_t}\mathbf{P}_{\delta\hat{\mathbf{x}}_t\delta\hat{\mathbf{x}}_t}\mathbf{H}^T_{x_t} + \mathbf{R} \tag{2.116}$$

This covariance represents a circular uncertainty over the measurements, and projection on to the 2-D image plane defines the region for some given number of standard deviations that the pixel measurement should fall within with high probability. Both [26] and [119] use this as a gating region for feature matching. Knowledge of this constraint in the feature correspondence search is expected to be known by the integrity algorithm.

2.12 INS/Image-Aiding Without Landmark States

Research by [28] and [87] represents two distinct approaches to image-aiding without the inclusion of the landmark positions in the state vector. Like methods in the previous discussion, these approaches assume that the primary purpose of the aiding algorithm is to limit INS error growth in relative navigation rather than maintain and update a map of an area. Both methods are described using a single, calibrated camera vision system, although extension to multiple cameras is possible.

The approach by [28] employs the extended Kalman filter estimating only the vehicle error states related to position, velocity, and accelerometer bias and scale. The focus of this approach is on estimating and removing translation error in the position estimate through the use of stochastic epipolar constraints. The authors assume that the selected gyroscopes can provide reasonable angular accuracy for several minutes, allowing gyro compensation for camera rotation, while avoiding use of the image data to improve the gyros. This assumption seemingly precludes the use of low-cost, lower quality INS, which has been shown to be one potential advantage of image-aiding [119]. However, the assumed accuracy of gyroscopes does reduce the complexity of the overall filter design by eliminating the attitude error states.

In [28], images are recorded from an initial time, t_a, to some subsequent time, t_b. The images are rotation-compensated using data from the gyros, prior to passing through a Harris corner detector [46]. The correlation process is accomplished using normalized cross-correlation like that earlier described by Equation (2.97). If a feature is persistent above a predetermined threshold, an observation vector for the m^{th} feature, $\mathbf{z}_m(t_i)$, is stored in a database along with the coordinates of the feature. The basic epipolar geometry is the same as that previously shown in Figure 2.13, but in [28], the constraint equation is now given based on the plane formed by a pair of observation vectors pointing at the same target and the translation between them:

$$(\mathbf{z}_m(t_a) \times \mathbf{z}_m(t_b)) \cdot \mathbf{p}_{ba} = 0 \tag{2.117}$$

where $\mathbf{z}_m(t_i)$ has replaced the s-vector terms and $\mathbf{p}_{ba}$ represents the translation occurring between times t_a and t_b. The form of this constraint equation is said to be coordinate-independent [41]. The cross product of the $\mathbf{z}_m(t_i)$ terms results in a vector perfectly orthogonal to the translation vector under ideal conditions, satisfying the constraint equation. However, errors in the vectors violate the constraint, thereby creating an observable residual error. Thus, the authors define a residual model of the form [28]:

$$\mathbf{r} = \left(\mathbf{I} - \mathbf{e}_x\mathbf{e}_x^T\right)\mathbf{e}_z\mathbf{e}_z^T\mathbf{p}_{ba} \overset{?}{=} \mathbf{0} \tag{2.118}$$

where

$$\mathbf{e}_x \equiv \frac{\mathbf{p}_{ba}}{\|\mathbf{p}_{ba}\|}$$

$$\mathbf{e}_z \equiv \frac{(\mathbf{z}_m(t_a) \times \ \mathbf{z}_m(t_b))}{\|(\mathbf{z}_m(t_a) \times \mathbf{z}_m(t_b))\|}$$

Although an explicit derivation of the residual model is not readily seen in either [28] or the related [27], this model is developed beginning with the basic definition of the vector difference of the cross product term and the projection of that vector onto the translation vector. This linear, least squares approach leads to the observation matrix, $\mathbf{H}$, used in the Kalman filter implementation [28]:

$$\mathbf{H} = \begin{bmatrix} \frac{\mathbf{r}^T}{\|\mathbf{r}\|} & \mathbf{0} \end{bmatrix} \tag{2.119}$$

This method is not without complications, recognizing that numerical instability may result if $\|\mathbf{r}\|$ or the denominators in $\mathbf{e}_x$ or $\mathbf{e}_z$ approach zero. However, a small residual value in the first case is an indication that the filter is performing well, and that a measurement should be included. As a result, the $\mathbf{H}$ matrix is redefined as [28]:

$$\mathbf{H} = \begin{bmatrix} \mathbf{e}_z^T & \mathbf{0} \end{bmatrix} \tag{2.120}$$

An empirically derived formula for the measurement variance is a notable element of [28]:

$$\sigma^2 \approx \frac{\overline{\mathbf{p}}_{ba}^2\sigma_a^2}{\|(\overline{\mathbf{z}}_m(t_a) \times \overline{\mathbf{z}}_m(t_b))\|^2} + \sigma_n^2 \tag{2.121}$$

where $\overline{(\cdot)}$ indicates that the value contains error, σ_a is the long-term deviation of the tracking algorithm, and σ_n is the standard deviation value for noise. The σ_n term is a nominal value, added to ensure that the overall variance is never zero, avoiding potential numerical problems in matrix inversion. Although not statistically rigorous, and clearly dependent on implementation elements such as imaging hardware, the tracking algorithm, vehicle trajectory, and observed scene, an explicit formulation for measurement variance is not normally shown in other papers cited in this dissertation. This method by [28] is still fundamentally based on observations from a pair of images, although the images may not necessarily be consecutive. The next method [87] proposes to use multiple images in performing image-aided updates.

The research by [87] uses landmark measurements to impose geometric constraints between multiple camera poses over time. The multiple image tracking requires camera measurements to be grouped per tracked feature, as opposed to aiding methods that apply constraints using only pairs of images and grouping camera measurements by camera pose, like that just described.

The vehicle error state vector, $\delta\hat{\mathbf{x}}_v$, structure for the Kalman filter is equivalent to that of [119], except that attitude is expressed in quaternions and the chosen reference frame is ECEF versus NED. The transformations between the image frame, camera frame, and real-world frame are also consistent with the equation structure shown in Section 2.9, noting that an additional DCM would be required to relate the ECEF and NED frames. The profound difference between the methods by [119] and [87] is that, in the latter, the error state vector is not augmented with target related states, $\delta\hat{\mathbf{x}}_t$, but rather states related to camera pose and attitude:

$$\delta\hat{\mathbf{x}}_c = \left[(\delta\hat{\boldsymbol{\psi}}_{c_1})^T \;\; (\delta\hat{\mathbf{p}}^e_{c_1})^T \;\; (\delta\hat{\boldsymbol{\psi}}_{c_2})^T \;\; (\delta\hat{\mathbf{p}}^e_{c_2})^T \;\; \ldots \;\; (\delta\hat{\boldsymbol{\psi}}_{c_n})^T \;\; (\delta\hat{\mathbf{p}}^e_{c_n})^T\right]^T \tag{2.122}$$

where the subscript C_i indicates the i^{th} camera pose, for $i = 1, 2, \ldots, N_{max}$, with N_{max} being a predetermined limit on the number of poses that can be used in the augmentation. The attitude error is described in three degrees-of-freedom [87]. The augmented error state vector is then represented by:

$$\delta\hat{\mathbf{x}} = \begin{bmatrix} \delta\hat{\mathbf{x}}_v \\ \delta\hat{\mathbf{x}}_c \end{bmatrix} \tag{2.123}$$

The IMU measurements are processed when they become available and are used to propagate the extended Kalman filter. The camera pose estimate is the sum of the IMU estimated position and known IMU-to-camera offset expressed in the ECEF frame. Each time an image is recorded, the current estimated camera pose information is added to the state vector, up to N_{max}. If the limit on the number of camera poses has been reached, older poses are replaced, but not before processing the feature observations. Since data is grouped in by feature, the image coordinates and camera frame feature position are indexed using the m^{th} feature in the i^{th} camera pose. Using the notation similar to that developed back in Section 2.9, the image measurement model is expressed as:

$$\mathbf{z}_i^{(m)} = \frac{1}{s_{z_j}^{c_i}} \boldsymbol{\pi} \mathbf{T}_c^{pix} \mathbf{s}_m^{c_i} + \mathbf{n}_i^{(m)} \tag{2.124}$$

where $\boldsymbol{\pi} = [\mathbf{I}_{2\times2} \;\; \mathbf{0}_{2\times1}]$. The indexed camera position is given by:

$$\mathbf{s}_m^{c_i} = \mathbf{C}_b^{c_i} \mathbf{C}_e^b \left(\mathbf{t}_m^e - \mathbf{p}_{c_i}^e \right) \tag{2.125}$$

If the target coordinates, $\mathbf{t}_m^e$, are not known a priori, they are estimated using a least-squares minimization based on the measurements for a specific feature and the filter estimates of the camera poses at the corresponding times [87]. If the target coordinates are known, the image measurements can be processed as soon as they are available. The

$\mathbf{H}$ matrix rows formed for each i, m pair of indices is remarkably similar to that shown in Equation (2.107), because the derivative of $\mathbf{h}$ with respect to the target, is found to be -1 times the derivative of $\mathbf{h}$ with respect to the camera pose. The fundamental contribution of this approach is the delayed linearization of the measurements, which makes it more robust to linearization errors.

*2.13 **GPS Integrity Methods***

This section describes the evolution of GPS integrity, beginning with key methods that formed the initial foundation in this field and moving toward the state-of-the-art in GPS integrity. The fundamental assumption in GPS integrity is that the number of measurements available is greater than the number of unknowns to determine, which in the case of GPS is normally four – three position states and one clock bias state. With regard to fault detection algorithms, a single redundant satellite measurement may be sufficient to detect a failure; however, mere detection is insufficient for integrity monitoring. Integrity requires at a minimum fault detection and isolation (FDI) capability, and in RNP more likely requires fault detection and exclusion (FDE) capability. According to [48], FDI requires at least two redundant satellites, with no more than one of the total number of satellites at any one time being affected by an undetected failure, while FDE requires even more redundant satellites under the same failure condition.

The detection of more than one simultaneous failure is a difficult analytical problem [25], and is currently a high interest research area [32; 69; 131], even though the probability of failure on multiple GPS satellites currently remains very low under normal (non-interference and unobstructed) conditions. However, this probability of failure is a function of the magnitude of error a navigation application can tolerate. In general, as the required degree of navigation accuracy is increased (i.e., the magnitude of error an application can tolerate is reduced), the probability of multiple failures is also expected to increase.

In [129], the authors categorize GPS integrity methods, first noting that primary GPS integrity methods can be categorized in two ways: (1) Snapshot methods, and (2) filtering methods. Snapshot methods are generally evaluated based on a single epoch only, with no memory of the previous history of measurements, using a least squares approach [48; 129]. The snapshot approach is used more frequently since it does not have to rely on receiver motion [48]. Filtering methods, on the other hand, typically parallel Kalman filter structures, with each filter based on a different hypothesis, even if integrity is evaluated on an epoch-by-epoch basis [129]. In [48], a filtering scheme describes using previous data in averaging or filtering, specifically the last estimates of position and possibly velocity to predict the incoming measurements.

In addition, [129] notes that these integrity algorithms can be further defined as either a position domain (based on observable position errors) or range domain (based on errors in pseudorange measurements) algorithm based on the characteristics of the test statistics used in FDE. In a least squares-based snapshot approach, the application the range domain and position domain methods are equivalent since there is a direct transformation between them. However, in the filtering approach, the relationship between the range domain test statistics and position domain errors in determining a horizontal protection limit is difficult to ascertain analytically when working in the range domain. This is due to the fact that the current navigation solution is conditioned on the past history of measurements and navigation system model. The evolution of the GPS integrity methods described in this section starts with snapshot methods and ends with filtering methods.

Both snapshot and filtering methods are extremely valuable, and the choice often depends on application and computational limitations, although the latter has become less of a restriction as software and hardware capability has grown significantly. A simple comparison is made in [58], stating that the snapshot method, although sometimes limited, is easier to implement, computationally less complex, and is more easily understood intuitively than filtering methods. However, the authors do recognize that the filtering method using Kalman filtering are expected to provide a higher level of performance based on the use of *a priori* information and the ability to incorporate the impact of the GPS receiver clock. Table 2.3 provides a simple summary of select GPS integrity methods discussed in the following sections.

2.13.1 ***Receiver Autonomous Integrity Monitoring***.

The appeal of GPS to both the military and civilian user community has led to a considerable amount of work in integrity algorithms being focused on this particular system [110]. Of special significance is Receiver Autonomous Integrity Monitoring (RAIM), which is useful because it is passive and localized to the individual GPS receiver, therefore not requiring a large and complicated infrastructure. Since the receiver lies at the end of the integrity processing chain, it is perhaps the most critical stage in integrity determination [79]. According to [90], RAIM algorithms are not standardized, but most current implementations are based on least-squares-based snapshot, or equivalent methods.

Three traditional integrity algorithms developed for RAIM have been found to largely form the foundation for integrity monitoring. Two of these algorithms are described in some detail in this section because of their importance, with the third evaluated in summary. Many newer algorithms have sought to extend on these basic concepts, and are discussed in later sections. As pointed out by [90], the basic approaches in GPS RAIM are limited in some degree in both the availability of detection and, especially, exclusion capability. This is due to limitations imposed by, for example, limited numbers of satellite measurements available, satellite geometry, and vulnerability to interference of both the GPS signal and signals from augmentation systems. The basic RAIM approaches are also restricted to a single measurement failure case [18; 24; 89]. It should be noted that the single failure assumption is largely valid for GPS, where RAIM is needed only to detect rare slow-growing failures that are expected to occur fewer than three times a year in GPS. In contrast to slow-growing errors, larger, nearly instantaneous, or “step,” failures commonly associated with hardware failure or other loss of measurement sources are more likely to occur, but also more likely to be detected. GPS receivers are often equipped to handle single, and potentially multiple, step failures.

Table 2.3: Summary Table: Select GPS Integrity Methods

Method	Classification	Description	Key References
Parity Vector	Snapshot, RAIM, position domain	Test statistic results from error transformation to the null-space of $\mathbf{H}^T$	[110] [117]
Residual	Snapshot, RAIM, position domain	Test statistic results from least squares residuals	[95]
Range Comparison	Snapshot, RAIM, range domain	Comparison of actual pseudorange measurements against alarm threshold	[68]
Slope	Snapshot, RAIM, range & position domains	Models linear relationship between horizontal position error and test statistic; HPL approximation based on simulated bias on satellite with least observable test statistic	[16] [18] [23] [130] [129]
Solution Separation	Snapshot/filtered, position domain	Kalman filtering used to compare position estimates between parallel filters	[10] [11] [20] [130] [129]
Extrapolation	Filtered, range domain	Kalman filtering used to compare residual statistics that are averaged in different cycle lengths	[6] [7] [8] [29] [30] [31]

2.13.2 ***Parity Vector.***

The parity vector approach to integrity monitoring was introduced by Potter and Suman [96], who first applied the methodology to inertial navigation. The approach was later reintroduced by Sturza [110] with regard to GPS applications, and is generally presented with more mathematical detail than the other foundational algorithms [16]. For this reason, a detailed derivation closely following that of [110] (unless otherwise noted) is shown here to provide reference when later drawing comparisons later with other algorithms.

The parity vector algorithm is predicated on the basis that the number of measurements, m, must exceed the number of dimensions, n, of the states being estimated by at least one, such that $m \geq n + 1$. The measurement model, linearized about a given state vector, **x**, is described by the equation:

$$\mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{w} + \mathbf{b} \tag{2.126}$$

where **z** is the $m \times 1$ resultant measurement vector based on the product of the $m \times n$ linear observation matrix, **H**, and the $n \times 1$ state vector, **x**, plus the $m \times 1$ vector of zero-mean Gaussian noise, **w**, with a diagonal covariance matrix, $\sigma^2\mathbf{I}_{m\times m}$. The final added vector, **b** ($m \times 1$), is a vector of zeros for the no fault condition, but gets some value b in the i^{th} element of **b**, representing a bias if a single fault occurs in the i^{th} measurement.

Since **H** is necessarily not a square matrix, but $m \times n$, let $\overline{\mathbf{H}} = \left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}^T$ represent the $n \times m$ Moore-Penrose pseudo-inverse of **H**, such that the least-squares estimate of **x** in state space is given by:

$$\hat{\mathbf{x}} = \overline{\mathbf{H}}\mathbf{z} = \left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}^T\left(\mathbf{H}\mathbf{x} + \mathbf{w} + \mathbf{b}\right) = \mathbf{x} + \overline{\mathbf{H}}\left(\mathbf{w} + \mathbf{b}\right) \tag{2.127}$$

In addition, the **H** matrix is assumed to be composed of linearly independent column vectors, allowing the decomposition of **H** into the product of a real orthonormal matrix **Q**

$(m \times m)$, such that $\mathbf{Q}^{-1} = \mathbf{Q}^T$, and an upper triangular matrix **R** $(m \times n)$, with the last $m - n$ rows of **R** containing only zeros [117]. Assuming for a moment that no noise or faults are present, the QR decomposition can be used to solve for **x** [4; 117; 129]:

$$\mathbf{Q}^T\mathbf{z} = \mathbf{R}\mathbf{x} \tag{2.128}$$

$$\begin{bmatrix} \mathbf{Q_x}^T \\ \mathbf{Q_p}^T \end{bmatrix} \mathbf{z} = \begin{bmatrix} \mathbf{U} \\ \mathbf{0} \end{bmatrix} \mathbf{x} \tag{2.129}$$

where **U** is the first n rows of **R** and $\mathbf{Q}^T$ is subdivided into matrices $\mathbf{Q_x}^T$ $(n \times m)$ and $\mathbf{Q_p}^T$ $((m-n) \times m)$ [117]. The matrix $\mathbf{Q_p}^T$ is also known as the parity matrix, **P** [18]. The rows of the parity matrix are orthogonal to **z**, thus the columns of **P** span the parity space of **H** [117]. As a result, when the measurements are corrupted by normally unobservable noise or biases, a transformation to the parity space can reveal their impact in the form of the parity vector, **p**, defined as:

$$\mathbf{p} = \mathbf{P}\mathbf{z} = \mathbf{P}\left(\mathbf{w} + \mathbf{b}\right) \tag{2.130}$$

with a distribution characterized as $\mathbf{p} \sim \mathcal{N}(\mathbf{Pb}, \sigma^2\mathbf{I}_{m-n})$. The $(m-n) \times 1$ parity vector is based on the assumptions that **p** and **x** are independent (valid only if **H** is known exactly [110]), and that the mean of **p** is zero under the no fault condition, but has a non-zero mean if a fault is present, allowing it to be used for fault detection.

A linear transformation matrix, **T** $(m \times m)$ can then be developed by augmenting the $\overline{\mathbf{H}}$ matrix, with the rank $m - n$ matrix **P** $((m-n) \times m)$, where the product $\mathbf{PP}^T = \mathbf{I}_{m \times m}$, and **P** is orthogonal to **H**, such that $\mathbf{PH} = \mathbf{0}$. Therefore, **T** maps the measurement space into the n dimensional state space and $m - n$ dimensional parity space.

Note that **T** is a full rank square matrix, such that the inverse matrix $\mathbf{T}^{-1}$ exists with $\mathbf{T}\mathbf{T}^{-1}$ $=\mathbf{T}^{-1}\mathbf{T} = \mathbf{I}$. Thus, $\mathbf{T}^{-1}$ is the linear transformation mapping back from the state space and parity space to the measurement space, allowing for the formation of a fault vector, **f** $(m \times 1)$:

$$\mathbf{f} = \mathbf{T^{-1}} \begin{bmatrix} \mathbf{0} \\ \mathbf{p} \end{bmatrix} = \left[\mathbf{H}|\,\mathbf{P}^T\right] \begin{bmatrix} \mathbf{0} \\ \mathbf{Pz} \end{bmatrix} = \left(\mathbf{P}^T\mathbf{P}\right)\mathbf{z} \tag{2.131}$$

where **f** is normally distributed with mean $\mathbf{P}^T\mathbf{Pb}$ and covariance $\sigma^2\mathbf{P}^T\mathbf{P}$. It should be noted that the matrix product $\mathbf{P}^T\mathbf{P}$ forms an idempotent matrix, such that this matrix times itself is equal to itself. Although **p** and **f** are defined in different spaces, their respective inner products yield the same results:

$$\mathbf{p}^T\mathbf{p} = (\mathbf{Pz})^T(\mathbf{Pz}) = \mathbf{z}^T\mathbf{P}^T\mathbf{Pz} \tag{2.132}$$

and

$$\mathbf{f}^T\mathbf{f} = \left(\mathbf{P}^T\mathbf{Pz}\right)^T\left(\mathbf{P}^T\mathbf{Pz}\right) = \mathbf{z}^T\mathbf{P}^T\mathbf{P}\mathbf{P}^T\mathbf{Pz} = \mathbf{z}^T\mathbf{P}^T\mathbf{IPz} = \mathbf{z}^T\mathbf{P}^T\mathbf{Pz} \tag{2.133}$$

thereby producing a single, non-negative scalar value that can be used as a test statistic, denoted as D, for fault detection. An evaluation of dual hypotheses $\mathbf{H}_0$, the case where no fault occurs, and $\mathbf{H}_1$, the condition under which a single fault is present, can then be given by:

$$D \underset{\mathbf{H}_0}{\overset{\mathbf{H}_1}{\gtrless}} \gamma \tag{2.134}$$

where γ is the threshold value for comparison, based on the false alarm rate probability, p_{fa}, number of redundant measurements, and covariance of the noise. If D = $\mathbf{p}^T\mathbf{p} = \|\mathbf{p}\|^2$

$> \gamma$, then a fault is declared. As described in [18], it is often more convenient to represent the decision space in terms of $D = \|\mathbf{p}\|$, and the monotonic relationship between $\|\mathbf{p}\|$ and $\|\mathbf{p}\|^2$ allows computations using chi-square statistics on $\|\mathbf{p}\|^2$ and then taking the square root to find the corresponding values for $\|\mathbf{p}\|$. Using the latter case as the test statistic, the detection threshold, γ, can be computed as [117]:

$$\gamma = \sigma\sqrt{2}\,\text{erfc}^{-1}(p_{fa}) \quad \text{for } m - n = 1 \tag{2.135}$$

and

$$\gamma = \sigma\sqrt{2}\,\text{erfc}^{-1}\left(\frac{p_{fa}}{m}\right) \quad \text{for } m - n \geq 2 \tag{2.136}$$

given that [117]:

$$p_{fa} = \text{erfc}\left(\frac{\gamma}{\sigma\sqrt{2}}\right) \tag{2.137}$$

where erfc is the complimentary error function [18; 117]:

$$\text{erfc(s)} = \frac{2}{\sqrt{\pi}}\int_s^\infty e^{-\xi^2} d\xi \tag{2.138}$$

Figure 2.14 illustrates the relationship between the no-fault case and the case where a bias is present. As with all cases when noise uncertainty is a factor, two conditions exist for misidentification of the system status: The case where D $> \gamma$, but no failure actually occurred (false alarm), $p_{fa} = P\left[D > \gamma | H_0\right]$, and the case where D $< \gamma$, but the system has actually failed (missed detection), $p_{md} = P\left[D < \gamma | H_1\right]$.

The derivation presented here was developed in general mathematical terms and, as shown by Sturza in [110], can be implemented in both GPS and INS systems.

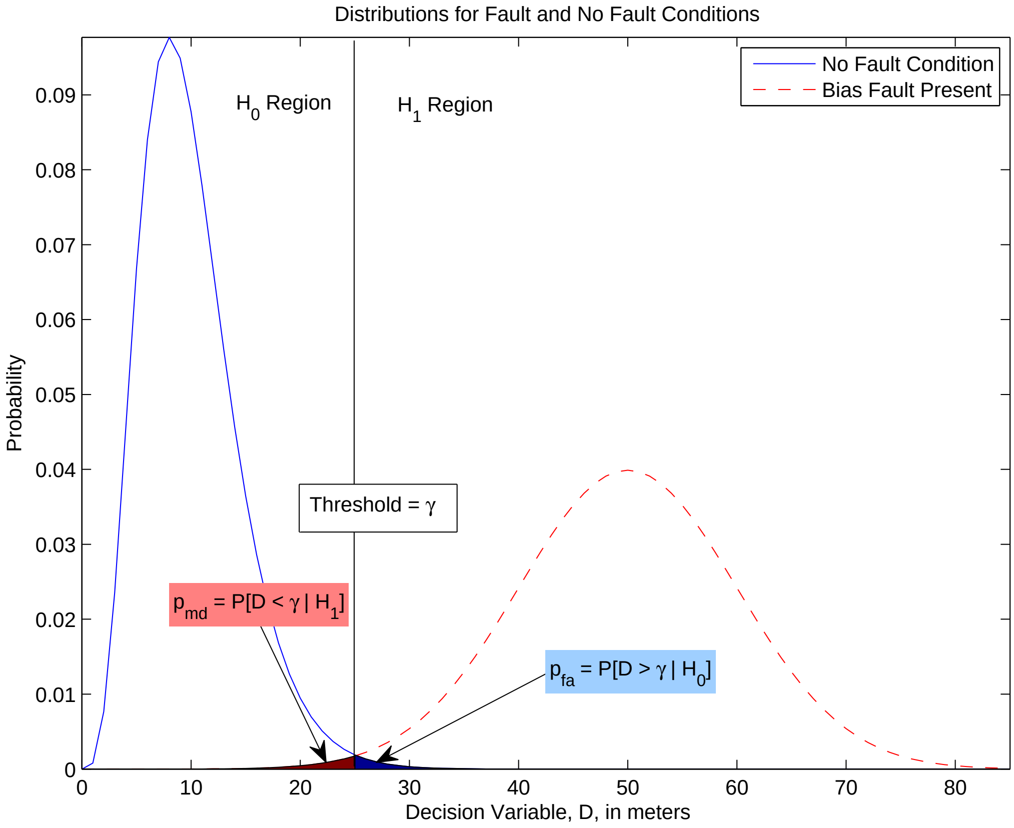

Figure 2.14: Illustration of Decision Regions for Fault Detection.

2.13.3 ***Least Squares Residual.*** Foundational work on the least squares residual concept applied to GPS integrity monitoring is commonly attributed [16; 39; 70; 79] to Parkinson and Axelrad [95], and is summarized here (see also Table 2.3). Although not explicitly stated in [95], under the general least squares assumption, the system is overdetermined, such that $m > n$, meaning there are $m - n$ degrees of freedom (DOF).

As in the parity vector case, fault detection depends on having at least one redundant measurement, such that $m \geq n + 1$. As stated earlier, the authors use $n = 4$, for the four unknowns in the position solution; three position coordinates and one clock bias term. In their original work, the measurement equation is given in terms of the pseudorange:

$$\rho_i = d_i - \left[\mathbf{e}_i^T 1\right] \mathbf{x} - \varepsilon_i \tag{2.139}$$

where ρ_i is the pseudorange to the i^{th} satellite, d_i is the projection of the vector from Earth center to the i^{th} satellite onto the user line of site to the i^{th} satellite, $\mathbf{e}_i$ is the unit vector along the user line of site to the i^{th} satellite, $\mathbf{x}$ is the n-length position vector, and ε_i is the normally distributed measurement error with mean μ_i and variance σ_i^2.

With the noted exception of the noise term, the vector representation of the measurements given by [95] can be arranged such that it is expressed in a form similar to that in Section 2.13.2 [95]:

$$\mathbf{z}_{m\times 1} = \mathbf{d}_{m\times 1} - \boldsymbol{\rho}_{m\times 1} = \begin{bmatrix} \mathbf{e}_i^T 1 \\ \cdots \\ \cdots \\ \mathbf{e}_m^T 1 \end{bmatrix} \mathbf{x}_{n\times 1} + \boldsymbol{\varepsilon}_{m\times 1} = \mathbf{H}_{m\times n}\mathbf{x}_{n\times 1} + \boldsymbol{\varepsilon}_{m\times 1} \tag{2.140}$$

and, as before, the least squares estimate can be expressed as:

$$\hat{\mathbf{x}} = \left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{z} = \overline{\mathbf{H}}\mathbf{z} \tag{2.141}$$

with an estimate of the measurement given by:

$$\hat{\mathbf{z}} = \mathbf{H}\hat{\mathbf{x}} = \mathbf{H}\overline{\mathbf{H}}\mathbf{z} \tag{2.142}$$

The vector of residual errors is formed by the difference between the predicted and actual measurements:

$$\hat{\boldsymbol{\varepsilon}} = \mathbf{z} - \hat{\mathbf{z}} = \mathbf{z} - \mathbf{H}\overline{\mathbf{H}}\mathbf{z} = \left(\mathbf{I} - \mathbf{H}\overline{\mathbf{H}}\right)\mathbf{z} \tag{2.143}$$

Note that $\mathbf{H}\overline{\mathbf{H}}$ is an $m \times m$ matrix that projects a vector onto column space of $\mathbf{H}$, while $\mathbf{I} - \mathbf{H}\overline{\mathbf{H}}$ is an $m \times m$ matrix that projects the vector on to the orthogonal complement of the column space, the left nullspace (i.e., the nullspace of $\mathbf{H}^T$). Replacing $\mathbf{z}$ with Equation (2.140) in the above result shows that the measurement error is also folded into the residuals:

$$\begin{aligned}
\hat{\boldsymbol{\varepsilon}} &= \left(\mathbf{I} - \mathbf{H}\overline{\mathbf{H}}\right)\left(\mathbf{H}\mathbf{x} + \boldsymbol{\varepsilon}\right) \\
&= \mathbf{H}\mathbf{x} - \mathbf{H}\overline{\mathbf{H}}\mathbf{H}\mathbf{x} + \boldsymbol{\varepsilon} - \mathbf{H}\overline{\mathbf{H}}\boldsymbol{\varepsilon} \\
&= \mathbf{H}\left(\mathbf{I} - \overline{\mathbf{H}}\mathbf{H}\right)\mathbf{x} + \left(\mathbf{I} - \mathbf{H}\overline{\mathbf{H}}\right)\boldsymbol{\varepsilon} \\
&= \mathbf{H}\left(\mathbf{I} - \left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{H}\right)\mathbf{x} + \left(\mathbf{I} - \mathbf{H}\overline{\mathbf{H}}\right)\boldsymbol{\varepsilon} \\
\hat{\boldsymbol{\varepsilon}} &= \mathbf{H}\left(\mathbf{I} - \mathbf{I}\right)\mathbf{x} + \left(\mathbf{I} - \mathbf{H}\overline{\mathbf{H}}\right)\boldsymbol{\varepsilon} = \left(\mathbf{I} - \mathbf{H}\overline{\mathbf{H}}\right)\boldsymbol{\varepsilon}
\end{aligned} \tag{2.144}$$

For the least squares problem, the average error in m equations is conveniently defined by the sum of the squares of the error [108]. The authors adopted this convention in coining the widely used sum-squared error (SSE) term for the sum of the squares of the range residual errors [95]. The SSE is then defined as the inner product of the residuals, $\hat{\varepsilon}^T\hat{\varepsilon}$, or, equivalently, as the trace of the outer product, trace$\left(\hat{\varepsilon}\hat{\varepsilon}^T\right)$.

As a non-negative scalar value based on the error, the SSE is useful as a test statistic. With ε drawn from independent identically-distributed (i.i.d) zero-mean Gaussian random variables, the sum of the squares of the errors are said to exhibit a chi-squared distribution, with $m - 4$ degrees of freedom in this case. The proposed chi-squared test statistic is then given by:

$$r = \sqrt{\frac{\hat{\varepsilon}^T\hat{\varepsilon}}{m-4}} = \sqrt{\frac{SSE}{m-4}} \tag{2.145}$$

The noncentral chi-squared distribution must be considered when the errors exhibit a non-zero mean, but according to the authors, the distribution is affected more by geometry than by the non-centrality parameter. The non-central chi-square density function cannot be written in closed form, but can be approximated using a finite number of terms and integrated numerically [18; 95]. In [18], the non-centrality parameter, λ, is proven to be:

$$\lambda = \left(\frac{pbias}{\sigma}\right)^2 \tag{2.146}$$

regardless of the number of degrees of freedom, and where $pbias$ is the magnitude of the deterministic bias portion of the pseudorange error for a satellite, as observed in the parity space, and σ is the standard deviation of the pseudorange measurement noise. This apparent cross-over between the parity method and least-squares residual method is not coincidental, and will be discussed in the next section.

The integrity check process for the least-squares residual method involves computing r for all satellites in view and comparing to some threshold, γ. The threshold values are generated through Monte Carlo analysis and selected according to false alarm and missed detection probability requirements. A failure is detected if $r > \gamma$. A failed satellite can potentially be identified by running the test against m subsets of $m - 1$ satellites.

2.13.4 ***Parity Vector & Residual Methods Commonality***. As it turns out, the $\mathbf{I} - \mathbf{H}\overline{\mathbf{H}}$ matrix presented in Section 2.13.3 is equivalent to the $\mathbf{P}^T\mathbf{P}$ matrix used to compute the fault vector, **f**, shown in Section 2.13.2, as $\mathbf{f} = \left(\mathbf{I} - \mathbf{H}\overline{\mathbf{H}}\right)\mathbf{z} = \mathbf{P}^T\mathbf{P}\mathbf{z}$. This consistency is reassuring given that the near optimal solution to the least-squares problem is found through maximum likelihood estimation, and we arrive at the same conclusion through these two different approaches.

Equivalence between these two methods was examined by Brown and Kelly in other notable papers [15; 16; 62] on GPS integrity monitoring. Brown drew an additional comparison to the "Range-Comparison Method" (RCM) proposed by Lee [68] (see Table 2.3), which uses the differences between five satellite pseudorange measurements and the respective predictions of the pseudoranges to determine an abnormal state. This method apparently actually predates the least squares algorithm given in [95]. The premise for Lee's method is that if all satellites are operating normally, the distribution of the errors will be zero mean, but if a single satellite is failing, then the subsets of the satellites excluding the failed satellite will have zero mean errors, but subsets including the failed satellite will not. This concept sounds similar to the parity vector approach, and indeed Brown showed that a linear transformation exists to take the RCM test statistic into the parity vector **p**.

Kelly [62] also evaluated the three methods considered by Brown and drew the same conclusion, believing that the three methods are essentially equivalent, provided an equal alarm rate being used to determine the threshold value. Kelly also considered the RAIM approach developed by Brenner [10] for an actual RAIM implementation tested by Honeywell, and found the method parallel to the parity vector approach, which is consistent with Brenner's own description. Brenner's method has also been called a solution separation method [8; 69] and will be discussed in detail later.

Ultimately, the parity vector and least squares residuals methods are core algorithms for RAIM, which is still considered the primary means of assuring GPS integrity today [55]. According to the previous discussions, the results of these methods should be comparable, but two conflicting opinions were seen as to which method should be implemented. Brown contends that the algorithm most easily implemented in real-time would appear to be the least-squares method [15]. However, Brenner states that the orthogonal transformations used in the parity-vector method preserve Gaussian statistics, and is thus a better implementation than models based on more complicated chi-square distributions [10]. The methods described here established a baseline for fault detection of bias-like ranging errors in navigation sensor measurements that provide the genesis for current and state-of-the-art integrity methods, as well as set a framework for the investigation of bearing errors as may occur in image-based sensors.

2.13.5 ***Evolution of RAIM.*** Initially, the integrity algorithms were able only to detect a failure, not necessarily identify the failing system. Adaptations of the algorithms and additional redundancies in the measurements have made possible not only the detection, but also the identification and possible exclusion of the source of bad measurements. Basic RAIM is restricted to the case of only a single measurement source failure. Selected key examples are discussed in this section to encapsulate the evolution of the integrity monitoring process.

An algorithm developed by [18] is based on the sum-square-of-the-residuals described in Section 2.13.3 as a test statistic. As described earlier, the test statistic has a chi-square distribution for analysis. In the proposed algorithm, a protection radius parameter is developed to approximate the true horizontal protection level under the worst case bias conditions [18]. The worst case bias is defined as the single satellite with the hardest-to-detect pseudorange bias, or bias that induces the largest missed alert probability when noise is present [18]. Although the algorithm can only approximate the true HPL, it serves very well to identify the satellites whose geometry negatively impact the geometric dilution of precision. The method in its basic form, however, can only be used for detection, not the joint detection and estimation problem [18]. The method is called the “Slope Method” in this dissertation. This method is especially useful for visualizing the impact of biased measurements on the position error.

The term “slope” is used to describe the ratio between the horizontal position error and the calculated test statistic. In other words, the slope for each satellite is a ramp-like model error trajectory that linearly approximates relationship between a growing pseudorange bias value and the impact of that bias on the horizontal position error. Using previously defined terms, the slope for the i^{th} satellite is given as the mapping function between the horizontal position space and the parity space [130]:

$$Slope_i = \frac{\sqrt{\overline{\mathbf{H}}_{1,i}^2 + \overline{\mathbf{H}}_{2,i}^2}}{\sqrt{\mathbf{S}_{i,i}}} \tag{2.147}$$

where $\mathbf{S} = \mathbf{P}^T\mathbf{P}$. Initial development of this method is based on the deterministic case (i.e., no noise is present) such that $\mathbf{p} = \mathbf{P}\left(\mathbf{b} + \mathbf{0}\ \right)$, and the horizontal position error resulting from a bias fault on the i^{th} measurement is given by [130]:

$$H_{\varepsilon_i} = Slope_i \left\|\mathbf{p}\right\| \tag{2.148}$$

Since the worst case bias condition is assumed, an initial upper bound on the position error is estimated by using the satellite associated with the largest slope [18], and is given by [130]:

$$\left|H_{bias}\right| = \max_{i=1:m} \left[Slope_i\ \right] \left\|\mathbf{p}\right\| \tag{2.149}$$

where $\max\left[Slope_i\ \right]$, $i = 1{:}m = 1, 2, ..., m$ is alternatively called $SLOPE_{\max}$. Of course, the measurements are not deterministic, and thus the process cannot reflect the true horizontal position error. However, statistically, if the measurement noise is assumed to be zero-mean AWGN, then the expected value of the parity vector is reasonably approximated by the deterministic case [110; 117]:

$$\begin{aligned} E\{\mathbf{p}\} &= E\{\mathbf{Pb}+\mathbf{v}\} \\ &= E\{\mathbf{Pb}\}+E\{\mathbf{v}\} \\ &= \mathbf{Pb}+\mathbf{0} \end{aligned} \tag{2.150}$$

The slope allows for a visualization of the projection of error into the observable parity space based test statistic in the form of a marginal density. Figure 2.15 shows an example of the slope relationship and the test statistic density using simulated data (3000 samples) for a bias error on satellite number 1, the satellite with the largest slope in this case. Figure 2.16 shows the same magnitude bias on satellite number 4, which has the second largest slope. From these illustrations, it easy to see how a bias on satellite 1 is harder to detect when looking at the distribution of the test statistic, and why this method considers using the maximum slope satellite in designing a protection radius. It should be noted, as pointed out by [91], and demonstrated in these figures, that a failure could occur on any of the available satellites, and thus the probability of an unacceptable solution could be overestimated and the RAIM availability underestimated.

A deterministic estimate of the total pseudorange error is insufficient in that it does not take into account the impact of the noise on the test statistic, as illustrated in the previous figures where the center of mass of the error ellipses (or approximately elliptical distributions) is representative of the deterministic bias. Given that this is a snapshot method, the realized value of the noise at any epoch may subtract or add to the range bias. In order to compensate for this uncertainty, and provide a better approximation of the upper bound, the horizontal protection radius is the $|H_{bias}|$ term given by Equation (4.10) computed using the $\|\mathbf{p}\|$ value that establishes the center of mass of the uncertainty ellipse at a point on the $SLOPE_{max}$ line where only an area equal to p_{md} is to the left of the threshold.

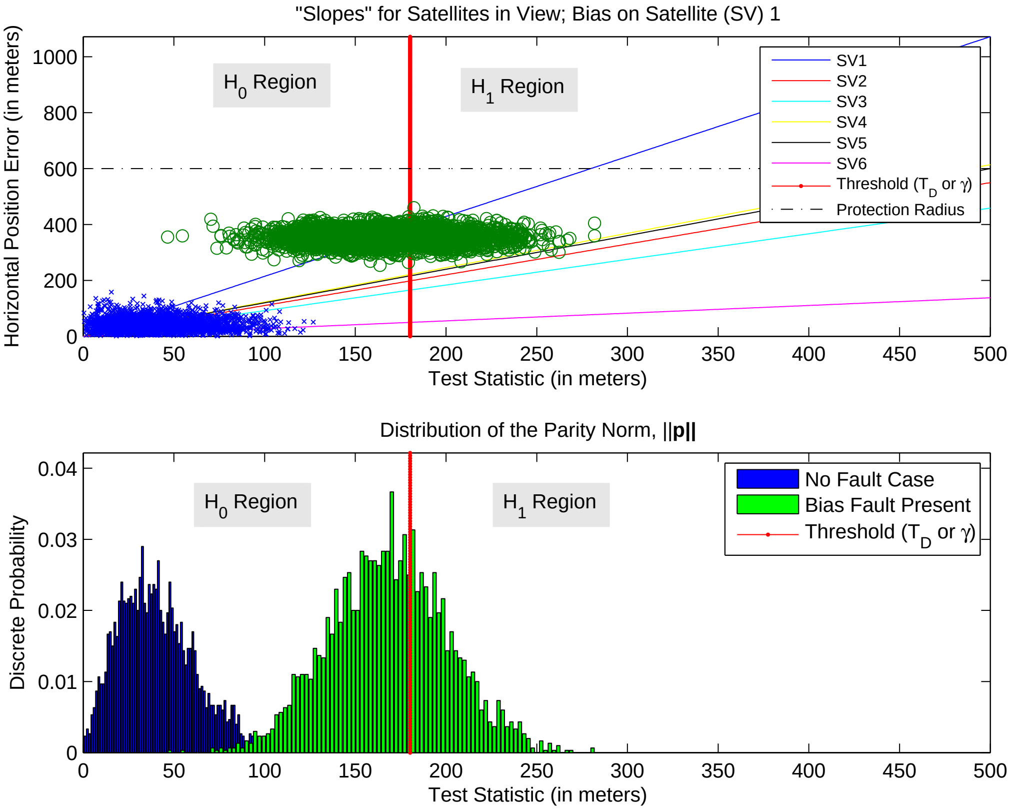

Figure 2.15: Impact of Bias on Satellite Number 1.

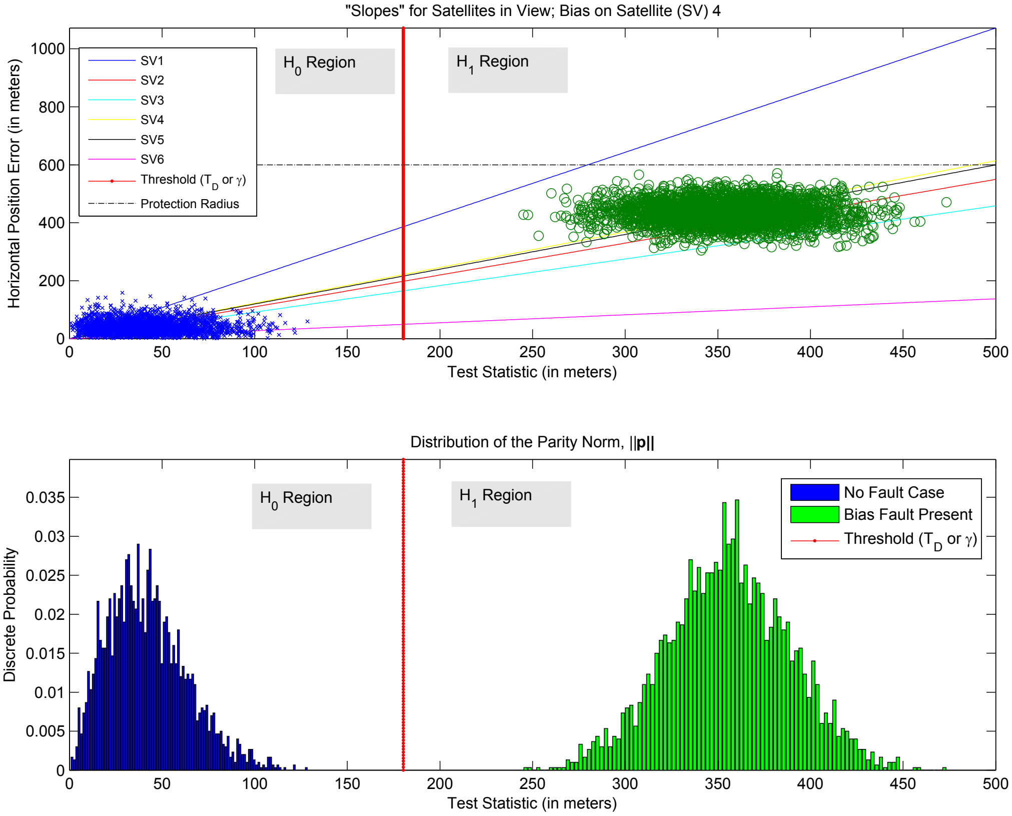

Figure 2.16: Impact of Bias on Satellite Number 4.

In the example illustrated in Figure 2.15 and Figure 2.16, this value is $\|\mathbf{p}\| = 279$. The protection radius is then represented by the horizontal line intersecting the horizontal position error (vertical) axis and the $SLOPE_{max}$ line at the given value of $\|\mathbf{p}\|$.

Another reference point in the evolution in RAIM is found in [98]. In this paper, Pullen, et. al., argue that the assumptions made in traditional RAIM algorithms do not consider any prior probability information that may influence false alarm and missed detection probabilities, and develop the idea of a prior probability model (PPM) based on [33]. The PPM provides information on whether or not potential failures exist before measurements are actually taken. Unlike in [33], where discrete failure parameters were used, this PPM used continuous Gamma and Normal distributions for spacecraft failures. Failure probability is based on the mean-time-to-restore (MTTR) and mean-time-between-failures (MTBF), and is given by the equation [33]:

$$P_F = \frac{MTTR}{MTTR + MTBF} \tag{2.151}$$

with each satellite evaluated against its individual P_F to determine how it would be modeled in terms of error in a scenario simulation.

The researchers then developed a cost function based on five possible RAIM results, including good position, detected bad position, missed detection, false alarm, and non-availability. The cost was minimized computing the cost for all possible discrete thresholds and choosing the threshold for the smallest cost for the point where the relative position error is exceeded. The idea introduces the idea of using cost-based models and a priori information. This paper set out to develop optimal thresholds for GPS geometries, but produced only mixed results [98]. However, their efforts showed that a Bayesian approach could prove useful, including a measurement history, which is the thrust of work by [94] and has later application in filtering methods.

In their paper [116], van Diggelen and A. Brown developed and proved two theorems related to GPS RAIM methods based on the least squares and parity vector methods.

They first showed that, given only one set of GPS measurements, that the fault vector is independent of the navigation error. Their mathematical proof follows that presented in Section 2.13.2, with an additional singular value decomposition of $\overline{\mathbf{H}}$ used in showing that the fault vector and navigation error are functions of two different independent, identically-distributed random variables, and thus themselves statistically independent of each other. This implies that it is essential to accumulate measurements over time before predicting navigation solution precision. Secondly, they showed that using the parity method of RAIM to correct a biased navigation solution is equivalent to removing the biased measurement in the first place, demonstrating the true strength of the parity method.

The mathematics detailed in [116] are described here to set up later discussions on filtering approaches to integrity. Building from notation defined in Section 2.13.2, a navigation solution difference term, $\mathbf{d}$, is defined as the difference between the position vector, $\mathbf{x}$, and the best estimate of $\mathbf{x}$, or $\hat{\mathbf{x}}$ [116]:

$$\mathbf{d} = \hat{\mathbf{x}} - \mathbf{x} = \overline{\mathbf{H}}\mathbf{e} \tag{2.152}$$

where $\mathbf{e}$ is the error in the measurements. Redundant measurement requirements allow standard RAIM techniques to estimate the measurement error $\mathbf{e}$, and thereby infer the value of the measurement error, $\mathbf{d}$. The estimate of $\mathbf{e}$ is commonly done through the use of the parity vector, $\mathbf{p}$, or fault vector, $\mathbf{f}$, as shown in [110], and described in detail earlier. The fact that both the parity, $\mathbf{P}$, and $\mathbf{S} = \mathbf{P}^T\mathbf{P}$ matrices are orthogonal to $\mathbf{H}$ allows $\mathbf{p}$ and $\mathbf{f}$ to be redefined as [116]:

$$\mathbf{p} = \mathbf{P}\mathbf{e} \tag{2.153}$$

and

$$\mathbf{f} = \mathbf{S}\mathbf{e} \tag{2.154}$$

A single bias is assumed in the i^{th} element of $\mathbf{e}$, which in turn manifests in the i^{th} element of $\mathbf{f}$, through the relationship [116]:

$$f_i = \mathbf{S}_{(:,i)}\mathbf{e} \tag{2.155}$$

where the i^{th} column of $\mathbf{S}$ is used to couple the bias error of $\mathbf{e}$ to f_i. Based on the assumption that the remaining elements of $\mathbf{e}$ are zero-mean noise, the best estimate of the error is given by [116]:

$$\hat{\mathbf{e}} = [0\,0\ldots0\;\hat{e}_i\;0\ldots0\,]^T \tag{2.156}$$

Substituting Equation (2.156) into Equation (2.155) results in the relationship [116]:

$$\hat{e}_i = \frac{f_i}{\mathbf{S}_{(i,i)}} \tag{2.157}$$

allowing the estimate of the navigation error, $\hat{\mathbf{d}}$, based on the estimate of the bias, ê [116]:

$$\hat{\mathbf{d}} = \overline{\mathbf{H}}\hat{\mathbf{e}} = \overline{\mathbf{H}}_{(:,i)}\hat{e}_i \tag{2.158}$$

where $(:,i)$ indicates the i^{th} column of $\overline{\mathbf{H}}$.

Given a redundant system $m > n$, using the standard measurement model for $\mathbf{z}$, the fault vector $\mathbf{f}$ from Equation (2.154), and the difference factor $\mathbf{d}$ from Equation (2.152), and provided the elements of $\mathbf{e}$ are i.i.d. random variables, the following proof by [116] demonstrates the independence of $\mathbf{f}$ and $\mathbf{d}$:

Let the singular value decomposition of $\mathbf{H}$ be described by [116]:

$$\mathbf{H} = \mathbf{U}\begin{bmatrix}\mathbf{\Sigma}\\ \mathbf{0}\end{bmatrix}\mathbf{V}^T \tag{2.159}$$

where $\mathbf{U}$ and $\mathbf{V}$ are square orthonormal matrices and $\mathbf{\Sigma}$ is a diagonal $n \times n$ matrix. From the decomposition of $\mathbf{H}$, the following relationships are shown in [116]:

$$\begin{aligned}
\mathbf{d} &= \overline{\mathbf{H}}\mathbf{e} \\
&= \left(\mathbf{V}\,[\Sigma \;\; \mathbf{0}\,]\,\mathbf{U}^T\mathbf{U}\mathbf{e}\begin{bmatrix}\mathbf{\Sigma} \\ \mathbf{0}\end{bmatrix}\mathbf{V}^T\right)^{-1}\mathbf{V}\,[\Sigma \;\; \mathbf{0}\,]\,\mathbf{U}^T\mathbf{e} \\
&= \left(\mathbf{V}\,\Sigma^{-2}\,\mathbf{V}^T\,\right)\mathbf{V}\,[\Sigma \;\; \mathbf{0}\,]\,\mathbf{U}^T\mathbf{e} \\
&= \mathbf{V}\left[\Sigma^{-1} \;\; \mathbf{0}\,\right]\mathbf{U}^T\mathbf{e}
\end{aligned} \tag{2.160}$$

$$\begin{aligned}
\mathbf{f} &= \mathbf{S}\mathbf{e} = \mathbf{I} - \mathbf{H}\overline{\mathbf{H}}\mathbf{e} \\
&= \mathbf{I} - \mathbf{U}\begin{bmatrix}\mathbf{\Sigma} \\ \mathbf{0}\end{bmatrix}\mathbf{V}^T\mathbf{V}\,[\Sigma \;\; \mathbf{0}\,]\,\mathbf{U}^T\mathbf{e} \\
&= \mathbf{I} - \mathbf{U}\begin{bmatrix}\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix}\mathbf{U}^T\mathbf{e} \\
&= \mathbf{U}\left[\mathbf{I} - \begin{bmatrix}\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix}\right]\mathbf{U}^T\mathbf{e} \\
&= \mathbf{U}\begin{bmatrix}\mathbf{0} \\ \mathbf{I}\end{bmatrix}[\mathbf{0} \;\; \mathbf{I}\,]\,\mathbf{U}^T\mathbf{e}
\end{aligned} \tag{2.161}$$

the final step the authors present to prove independence is to define the transformed error estimate, ẽ such that [116]:

$$
\begin{aligned}
\tilde{\mathbf{e}} &= \begin{bmatrix} \tilde{\mathbf{e}}_1 \\ \tilde{\mathbf{e}}_2 \end{bmatrix} \\
&= \mathbf{U}^T \mathbf{e}
\end{aligned}
\tag{2.162}
$$

with $\tilde{\mathbf{e}}_1 \in \mathbb{R}^n$ and $\tilde{\mathbf{e}}_2 \in \mathbb{R}^{m-n}$. As a result, the expressions for $\mathbf{d}$ and $\mathbf{f}$ can be rewritten in terms of the elements of $\tilde{\mathbf{e}}$ [116]:

$$
\mathbf{d} = \mathbf{V}\Sigma^{-1}\tilde{\mathbf{e}}_1 \tag{2.163}
$$

$$
\mathbf{f} = \mathbf{U} \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} \tilde{\mathbf{e}}_2 \tag{2.164}
$$

and therefore, $\mathbf{d}$ and $\mathbf{f}$ are functions of n and $m - n$ different random variables, respectively, under the assumption the $\mathbf{e}$ is made up of i.i.d random variables. As a result of the analysis done by [116], a significant case is made for filtering approaches to integrity methods where possible. This is based on the fact the authors have shown that, provided only one set of GPS measurements, it is impossible to determine the navigation error from the fault vector due to their independence from each other. This work demonstrated that multiple epochs improve results and relates accuracy to integrity. Later discussions describe why use of multiple epochs may not be possible from a straight forward RAIM approach. Filtering approaches would obviously help in tracking changes of what are shown here to be random variables, while [39] also adds that the thresholds are themselves random variables. An additional proof in [116], shows that whether the bias error is corrected or excluded before measurement incorporation the influence on the navigation solution is the same. The exclusion is accomplished by zeroing the i^{th} row of the $\mathbf{H}$ matrix before performing the pseudoinverse and subsequent operations. This methodology is relevant to the method by [11], described in detail later.

In more recent, related work, Arthur [4] investigated the independence of the fault vector and navigation solution precision, without assuming that only one measurement contains a bias. At the same time he examined the more difficult case where biases are on the same order as the variance. Arthur recognized that although $\mathbf{p}^T\mathbf{p} = \mathbf{f}^T\mathbf{f}$ from the parity vector method, the dimensions of $\mathbf{p}$ and $\mathbf{f}$ differed, causing part of the error to be unobservable in the parity space. He proposes the concept of a "Gamma-space" and introduces a scaling matrix created to improve upon previous techniques in calculating confidence levels of position error magnitude. This method also potentially handles cases where more than one bias is present (i.e., multiple failures).

Recently, an investigation was done into the least squares approach presented earlier, where it was assessed to be limited in potential for multiple failure detection due to loss of information through the projection to the subspace [79]. This loss of information in the multiple failure case results from mutual cancelation of the errors. The proposed technique attempts to reconstruct lost information through the use of measurements collected over multiple epochs.

The initial mathematical development parallels that found in Section 2.13.3. However, in this case, resultant projections of the error vector are collected, epoch to epoch, taking advantage of the fact that the geometry between the user and satellite changes during these periods. In this way, the algorithm monitors each line of site and the norm of the residual vector, in order to reconstruct the error vector itself. One difficulty faced in this method is that the assumption of independence between the subspaces between epochs cannot be met due the constant clock offset value in the $\mathbf{H}$ matrix. Thus, the process evaluates errors in the subspace orthogonal to both the nullspace and last column of $\mathbf{H}$. Another problem is that with only small changes in relative geometry, ill-conditioned matrices may be produced, complicating the least squares computation.

This section fully described the fundamental RAIM algorithms providing a foundation for current and future work. The basic equivalence of the parity vector method and least squares method was clearly shown, and relevant extensions of the base theory was investigated. In current literature, the terms parity vector and least squares are often used synonymously, although the math expressed may reflect one particular approach or the other.

2.14 ***Recent Directions in Integrity Monitoring***

As described previously, RAIM was initially developed during a timeframe where GPS was considered only as an independent system. At that time, GPS was not the primary navigation system for flight, and selective availability (SA) was the dominant error source. Now, SA has been turned off, and GPS has become a mainstay of navigation. This has led to new challenges in and increased demands on integrity monitoring capability.

2.14.1 ***Weighting Approaches**.* In the absence of SA, other satellite error sources must be considered, some of which are unique to each satellite [93]. Although developed while SA was still a factor, integrity approaches presented by van Graas, et al, [117] and Walter, et al, [124] have application today. The methods extend the least squares (or equivalently, parity vector) methodology to apply weighting to the relative satellite errors by multiplying by a measurement covariance matrix before deriving the parity vector. This allows accounting for cases where the measurements from the satellites are not assumed to have the same statistics.

A newer method, called *NIO*RAIM has been proposed by Hwang and Brown [54] to apply non-uniform weighting to the measurements in an effort to better balance position accuracy with integrity monitoring. In fact, the authors propose increasing integrity availability at the expense of accuracy (while still meeting accuracy requirements), building on the concepts and detailed mathematics introduced by Ober [92]. The method itself is based in the least squares method and also incorporates the SLOPE factor shown earlier.

A reduction of the maximum slope implies that RAIM availability would be improved through reducing the horizontal protection limit. This is done via weighting the individual satellite measurements. An optimal method for determining the required weights has not been found, so an ad-hoc iterative method is applied. Extensive Monte Carlo analysis is done to generate table of look up values to be used in real-time.

Another recent proposal is the Operationally Weighted Average Solution (OWAS, renamed OWAS-1 for single fault detection and OWAS-2 for working with two different satellite constellations) presented by Lee [71]. Like the *NIO*RAIM concept, the OWAS approach seeks to improve RAIM availability, with the recognized trade off between position accuracy and integrity. The key difference between the two methods is that *NIO*RAIM applies numerically-derived weighting to the range measurements while OWAS-1 derives weights analytically, and in the position domain, to obtain a position solution [71]. OWAS-2 proposes use of the weighted average of two individual GPS and Galileo solutions, again applying the OWAS-1 weighting concept in the position domain. OWAS-2 is also intended to handle multiple faults.

2.14.2 ***GPS Integrity Channel (GIC)***. For a GPS-only system there is no independent method for verifying whether a detection is false or correct [93]; therefore, a number of ground-based and space-based augmentation systems have been proposed to improve the integrity associated with GPS. These include such systems as the Local Area Augmentation System (LAAS) [44], Wide Area Augmentation System (WAAS) [103], European Geostationary Navigation Overlay System (EGNOS) [118], and the Japanese Multi-Functional Satellite Augmentation System (MSAS) [102], all designed to monitor GPS performance real-time.

These systems relay error information (atmospheric error data for example) and/or differential corrections to GPS users, which allows for updating of position estimates, improving user accuracy. Information is also provided regarding the health status of satellites, which aids in integrity monitoring in the user GPS receiver. In the event of satellite failure, a “don’t use” message indicating to the user receiver that a particular satellite should be removed from the navigation solution, thereby improving integrity for the user. However, RAIM is still done by the individual receivers, commonly using methods previously described.

These augmentation systems are all considered RF-based, and thus vulnerable to the same RF interfence/jamming vulnerabilities as the original GPS signal. This is also true, although perhaps to a lesser extent with multiple frequencies, for methods that propose using multi-constellations like GPS-GLONASS-Galileo-QZSS. As extensive research in this area is ongoing, it is worth mentioning here for completeness. However, these augmentation systems are peripheral to the area of research outlined in this dissertation.

2.15 ***Integrated Systems***

The fundamental RAIM algorithms based on GPS-only measurements require that there are more measurements available than the number of unknowns in the position vector, i.e. $m \geq n + 1$. This condition cannot always be met, therefore it makes intuitive sense that improvements in integrity would be derived from additional independent measurements, even when using instantaneous measurements. As discussed earlier, Potter and Suman [96] explicitly showed that the parity vector algorithm could be applied to other systems, such as INS (prior to the adaptation to GPS by Sturza [110]), while the least-squares algorithm is generally applicable to any overdetermined system, which reemphasizes the mathematical detail provided in these sections. Consequently, these methods are being extended to the opportunity provided by multi-constellations of GNSS satellites and GPS integrated systems.

2.15.1 ***Multi-Constellations****.* With GLONASS already in existence, and the potential for new navigation satellite systems such as Europe’s Galileo System, Japan’s Quazi-Zenith Satellite System (QZSS), and China’s Beidou Navigation System (BNS), there has been research into the potential for improving GPS integrity through the use of multi-constellations. Most recent work has focused on the combination of GPS with Galileo because of Galileo's near-term operational status and the fact that these two particular constellations will provide greater global coverage than the others.

Because of similarities in position determination from space based platforms, it appears in large part, that the integrity algorithms being developed primarily extend the GPS RAIM parity vector/least squares algorithms previously described to the broader class of GNSS, and include the use of weighting proposed by [69; 71]. The anticipated advantage of using multi-constellation is obviously having more independent measurements available to help determine position as well as having different frequencies available to help mitigate atmospheric errors. In terms of integrity, the additional measurements can help with detection and exclusion of multiple failures. Realizing that there may be novel approaches still to be uncovered, the vulnerability to RF interference is still a potential problem for these systems.

2.15.2 ***Multi-Sensor Integration****.* In an effort to produce a consistent navigation solution during periods of GPS outages, GPS receivers have long been integrated with a large variety of different, already existing, navigation systems. Baro-aiding, integration with inertial navigation systems (INS), tighter integration with inertial measurement units (IMU), and integration with ground based radiofrequency (RF) location systems (e.g. Loran and pseudolites) are just a few examples. Integrity has always been a consideration, but is not often investigated in detail in publications on the subject. The overarching principle, once again, is that integrity improvement is expected when more independent measurements are available. This section provides some examples of integration schemes and the chosen approach to integrity monitoring.

The integration of GPS with the Loran systems has been under consideration for more than a decade. In [36], Enge, et. al., proposed modeling the Loran measurement equation as a pseudorange given by [36]:

$$\rho_i = r_i + c(b + g_i) + cd_i \tag{2.165}$$

where ρ_i is the Loran pseudorange, r_i is the actual Loran path length, c is the speed of light, b is the receiver clock offset from GPS time, d_i is the propagation delay, and g_i is the unknown offset between Loran and GPS. This form is then similar to (Equation (2.139)), and can thus be incorporated into the least squares model used for RAIM.

More recent approaches examine integration with Loran and include INS as well. In these cases, the IMU is typically integrated with GPS and loosely coupled with Loran. An example of this approach is given in [2], where the problem with TDOA measurements applied to RAIM was examined. When TDOA measurements are used, a single "master fault" appears as multiple faults to the receiver, violating the single fault assumption of RAIM. To counter this problem, the parity space matrix, **P**, was augmented by adding characteristic vectors for each Loran master station. Through this approach, the standard RAIM parity vector approach could be used.

2.15.3 ***Extrapolation Method.*** GPS is commonly integrated with some form of inertial system, with or without other additional systems such as Loran. Diesel, et al, [30] proposed an integrated GPS and inertial reference system (IRS) to provide WAAS specified integrity without the use of WAAS. This mechanism was designated the Autonomous Integrity Monitored Extrapolation (AIME$^{®}$) approach, simply called the extrapolation method in this dissertation (see Table 2.3). Since an aircraft is non-stationary, there are often periods of time where an insufficient number of satellites are in view to perform RAIM failure detection and exclusion. This problem could possibly be countered were the vehicle stationary, allowing the generation of more equations than unknowns through accumulation of measurements over time [30], but such is not the case here. Therefore independent RAIM "snapshots" must be used to determine integrity.

However, the authors conclude that by replacing the dynamics model of the aircraft with the dynamics of the IRS errors, then the least squares approach of the smaller dynamics over extended time could effectively detect very small satellite clock errors using only four satellites in view. The integration is done using an open loop Kalman filter to update an error state vector. Separate Kalman filters are used to test the validity of the individual satellite measurements. The test statistics for RAIM are generated from the Kalman filter residuals based on a weighted average over time. There are actually three time averages computed, 2.5 minutes, 10 minutes, and 30 minutes. The statistics of these measures are thought to exhibit central chi-square distributions for the no fault case and non-central chi-square distributions for the fault case, as was described in Section 2.13.3.

The extrapolation method is one of the state-of-the art methods in INS/GPS-aided integrity investigated in this research to deepen understanding of integrity concepts. Chapter III provides greater detail of this method and results and analysis of initial simulation using GPS measurements.

2.15.4 ***Solution Separation.*** Another integrity method for INS/GPS integration under recent consideration is the solution separation (SS) by Brenner [10] (see Table 2.3) mentioned in Section 2.13.4 [8; 20; 72]. Recall that the method was based on the parity vector method with Gaussian assumptions of zero mean for the no fault case and a non-zero mean for the fault case. The algorithm uses a hierarchical tree, first examining the solution using all satellites, then removing one satellite measurement at a time, recomputing the solution, and comparing the new solution to the full solution [8]. This particular method uses only the instantaneous measurements. Solutions are compared to a threshold designed based on the required false alarm rate.

The solution separation method is another state-of-the art method studied for this research. Mathematical detail, along with results and analysis of initial simulation using GPS measurements is provided in Chapter III and comparisons are drawn with the extrapolation method and slope method.

In recent papers, Bhatti compares the SS and extrapolation methods, and recognizes that neither method makes provisions for detecting the error rate and, therefore, seeks to enhance the methods to do so [6; 7; 8]. Error rate identification would aid the error detection algorithms in meeting integrity requirements. High error rates quickly violate the threshold and are easily detected, but slow growing errors, like those associated with GPS clock drift and INS, are more troublesome. By using a snapshot of the measurements, the SS method is more likely to catch the rapidly changing errors, while the time averaging in the extrapolation method would likely aid in discovering the slower growth errors, but the detection must be done within an acceptable time-to-alarm. The author proposes an algorithm based on the detection of the rate of the conventional test statistic. This algorithm is implemented by modeling three new states in the Kalman filter, including one each for velocity, acceleration, and jerk. This allows for a detection of the change in the rate of the residuals. This state-of-the-art method is an extension of the extrapolation method in its implementation and is referenced, but not fully implemented, in the analysis in the next chapter.

2.16 ***Chapter Conclusion***

This chapter developed the necessary background for understanding the large number of topics involved in INS/GPS-aided and INS/image-aided navigation. The concepts of GPS integrity were explored to set a baseline for the development of an INS/image-aided integrity scheme. The concept of GPS integrity is well understood, but efforts continue to improve capabilities. It was shown that the fundamental RAIM algorithms developed early in the GPS era still form the core of today's GPS integrity algorithms. The methodology and basic principles of GPS integrity have remained consistent over time with the most recent advances coming through augmentation with additional ground-based or space-based systems, or through integration with INS systems. Considering these implementations is useful, but there are fundamental differences between the observations made in GPS and those made in image-based navigation, which drives the pursuit of a new algorithm for image-aiding that provides the same assurances required by GPS-based systems.

III. Simulation of Select Methods

This chapter presents more mathematical detail and simulation results from a basic implementation of the solution separation (SS) and extrapolation methods introduced in Chapter II.

3.1 Introduction

Simulations were developed and run for the solution separation and extrapolation methods to deepen understanding in both the underlying principles of navigation integrity and the specific intricacies of these two integrity algorithms. The project primarily focused on these two integrity methods for GPS/INS integrated systems, but for a comparative baseline, the RAIM method developed by [15; 18] was also implemented.

3.1.1 ***Simulation Background.*** The computer code for the implementation of the integrity algorithms was written in Matlab, using the tightly-coupled GPS/INS integration model developed in [119] as a foundation. Parallel Kalman filters were designed for the Brenner and Diesel algorithms, and run in an open loop configuration. The filters used an 18-state error model, including three each position, velocity, attitude states, accelerometer bias, and gyro bias states, as well as one baro bias state and two GPS clock error states. Preliminary work was done using a "perfect" IMU, and error-free baro and GPS measurements. Additional simulations implement a 100Hz HG-1700 IMU model, with perfect baro and GPS measurements. The noise-free measurements allow clearer analysis of the effects of intentionally induced pseudorange errors. The simulated aircraft trajectory involved a simple racetrack flight pattern, as shown in Figure 3.1.

The total run time of the simulated flight path is 1200 seconds for this initial implementation. In the efforts of both Brenner and Diesel, Selective Availability (SA) was turned on, requiring a delayed update cycle of 2-2.5 minutes to mitigate the correlation effects of SA on the GPS measurements. SA was not initiated in this simulation, so the

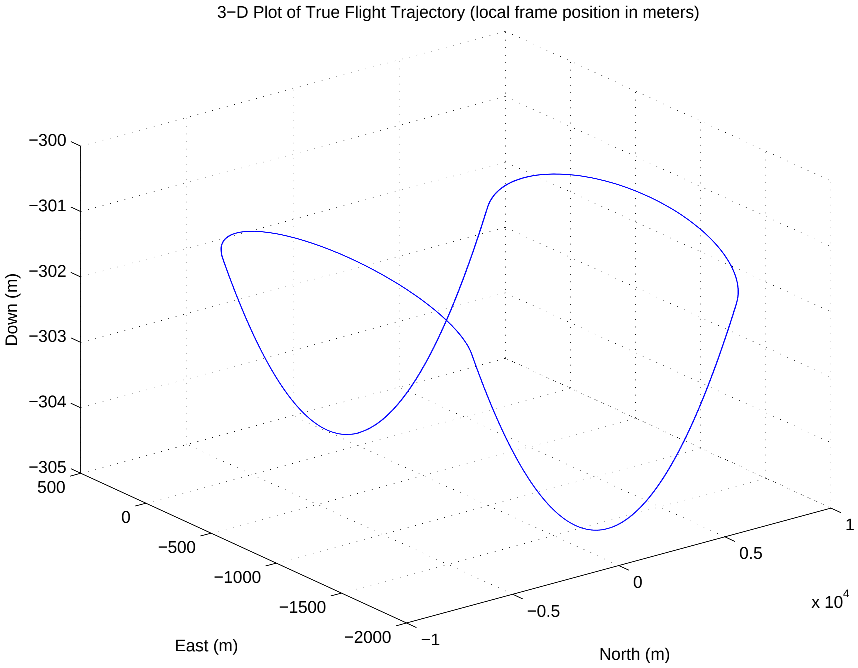

Figure 3.1: True Trajectory in Three Dimensions.

updates were run on an epoch-by-epoch basis, assuming perfect measurements in the absence of intentional error.

Integrity horizontal alert levels (HAL) specify the maximum allowable position error for a particular phase of flight. The non-precision approach phase (NPA) was chosen for reference in these simulations. This choice requires the horizontal protection limit (HPL), or maximum horizontal error that can be consistently detected with the specified probability of false alarm (p_{fa}) and probability of missed detection (p_{md}), to fall below an HAL of 0.3 nmi (or 555.6 m). The commonly used probabilities of false alarm and missed detection were set at $3.33e^{-7}$ and $1e^{-3}$, respectively, for all simulations [18]. In general, these preliminary simulations focused on error detection, not necessarily both detection and exclusion, although the potential for both is discussed.

3.2 Analysis

A RAIM method proposed by Brown [18] was investigated as a reference baseline for comparison with the later methods. This method is based on GPS only, unlike the

Brenner and Diesel methods, which involve integrated GPS/INS systems. In addition, the Brown RAIM method is "snapshot" method, relying only on the current measurements, while the Brenner and Diesel methods use the information from the Kalman filter based on measurement history. As stated in summary of [18], the protection radius determined by this method does not perfectly measure the HPL, but provides a reasonable approximation, especially under conditions of poor satellite geometries. This was clearly observed during the course of the simulations.

This RAIM method implements the parity vector approach deriving its information from the from the Kalman filter measurement observation matrix, $\mathbf{H}$, as was shown in Chapter II. Recall that the protection radius is then developed by taking the worst case "slope" of the satellite measurements used, and multiplying this slope by a normalized bias term based on $n-4$ degrees of freedom (DOF) from a non-central chi-square distribution based on the given probabilities for p_{fa} and p_{md} (note: 1-DOF determined by Gaussian methods [18]), as shown [18]:

$$\text{PROTECTION RADIUS} = (SLOPE_{\max})\sigma_{pr}(p_{bias}) \tag{3.1}$$

where σ_{pr} is the pseudorange measurement error and pbias is the normalized bias term. The bias terms used in this simulation, for both the Brown and Diesel methods, are based on Table 3.1 [18] for $p_{fa} = 3.33e^{-7}$ and $p_{md} = 1e^{-3}$. This parity vector times itself transpose is used as the test statistic in this method:

$$\text{Test statistic} = \sqrt{\mathbf{p}^T\mathbf{p}} = \|\mathbf{p}\| \tag{3.2}$$

A failure is declared if the test statistic exceeds the determined threshold calculated by the product of the normalized chi-square value from Table 3.1 and the σ_{pr} value:

$$\text{Detection Threshold} = (\sigma_{pr})(T_D) \tag{3.3}$$

Table 3.1: Non-Central Chi Square Parameters for Simulation

Chi Square DOF	Normalized bias term (p_{bias})	Normalized Chi Square Threshold (T_D)
1	8.194*	5.1037*
2	8.479	5.4615
3	8.687	5.7384
4	8.860	5.9750
5	9.009	6.1861
6	9.142	6.3789
7	9.263	6.5576
8	9.375	6.7251

*Actually determined by Gaussian Method

Unless otherwise stated, the $\sigma_{pr} = 2$ meters in the simulations since the measurements were generated noise-free in order to better assess the impact of induced errors.

3.2.1 ***Comparison with Solution Separation Method.*** The solution separation method is implemented with a bank of $n + 1$ Kalman filters, where n is the number of measurements. The full filter is the baseline filter, using all n satellite measurements, and denoted alternately as the full filter or as subfilter 0, as there are $n = 0$ measurements removed. In addition to the full filter, there are n subfilters, each having a single measurement excluded, identified by the index of the satellite measurement removed. For example, subfilter 2 is the subfilter that processes all measurements except the second one. The underlying assumption in the solution separation method is that in the case of a single failure, resulting in an incorrect measurement, the subfilter excluding that measurement will be unaffected. This condition allows for the detection of the failure, as the horizontal positions determined by the full filter and that of the subfilter excluding the bad measurement will diverge. This method is superior to attempting to explicitly model the errors, because if the error model does not sufficiently describe the error condition, or the nature of the error changes, filter performance will likely suffer and the overall performance of the navigation system may fail to meet required specifications.

The test statistic in the solution separation method is the root sum square of the differences between the horizontal position errors (in the local frame) as determined by the full filter and subfilter n:

$$\text{SS Test Statistic} = \sqrt{(\Delta x_0(1) - \Delta x_n(1))^2 + (\Delta x_0(2) - \Delta x_n(2))^2} \tag{3.4}$$

where the indices, 1 and 2, reflect the error states for the north and east position errors, respectively. Evaluation of the test statistic is done with regard to the covariance of the solution separation. In Brenner [11; 12], this covariance is propagated and updated using a process supplemental to the Kalman filter process. Equivalence between this process and the simple differencing of the covariance matrices produced by the Kalman filters was proven in [129]. As a result, this implementation of the Brenner algorithm does not include separate covariance propagation for the solution separation statistics, instead using information already provided by the parallel Kalman filters.

The threshold for error detection is formed by the square root of the largest eigenvalue, λ_{D_n}, of the horizontal position state covariance found from the solution separation covariance matrix (approximated by the matrix itself) multiplied by the scale factor related to the p_{fa} [11; 12]:

$$\text{Detection Threshold} = D_n = \sqrt{\lambda_{D_n}} Q^{-1}(p_{fa}/(2n)) \tag{3.5}$$

where $Q(s) = \dfrac{1}{\sqrt{2\pi}} \displaystyle\int_s^\infty e^{\frac{-t^2}{2}} dt.$

For the simulation, the test statistic is normalized by the detection threshold, such that a failure is declared when the maximum normalized test statistic of the n subfilters exceeds a value of 1. The HPL (denoted as H_{cov}) is determined using the detection threshold in conjunction with consideration for the covariance of the subfilter itself. The additional term is computed using the square root of largest eigenvalue, λ_{a_n}, of the horizontal position state subfilter n covariance (again approximated by the actual value) and scaling by a factor based on the p_{md} [11; 12]:

$$a_n = \sqrt{\lambda_{a_n}} Q^{-1}(p_{md}) \tag{3.6}$$

The HPL, or H$_{cov}$, is then equal to $max\{D_n + a_n\}$, over all subfilters, where n indicates the excluded measurement. Comparison plots between the basic RAIM method and solution separation method are shown in the following figures. In the first comparison, a total of seven satellites were used over a period of 200 seconds of flight time. There were no errors induced, and a perfect IMU and perfect measurements were used. Figure 3.2 shows the horizontal protection limits for the RAIM method and the solution separation method. As discussed earlier, the RAIM method is based on GPS, while the solution separation method takes advantage of the INS position solution as well. It is clear from Figure 3.2, that the protection level afforded the RAIM method is far more conservative than that of the solution separation method. The shape of the RAIM protection level appears to show the effects of changing satellite geometry with respect to the flight path, an artifact discussed earlier. Since no errors were induced, the test statistics for this case were very nearly zero, as shown in Figure 3.3, with the dotted line indicating the respective detection thresholds. For RAIM, with the pseudorange standard deviation of 2 and three degrees of freedom, the threshold is at 11.48, as described earlier, while the normalized threshold for the solution separation is one. This scenario was then run using only five satellites in view.

Figure 3.4 shows the results of the HPL evaluations when the number of satellites was reduced from 7 to 5. Notice that for both methods the level of the HPL has increased, reflecting the reduced number of redundant satellites. In the case of standard RAIM, this is the minimum number of satellites permissible. Based on the slope formula, it is possible to experience numerical difficulty under poor satellite geometry, as diagonal elements of the S matrix get very small. The integrated system shows slight rise in the HPL in the region

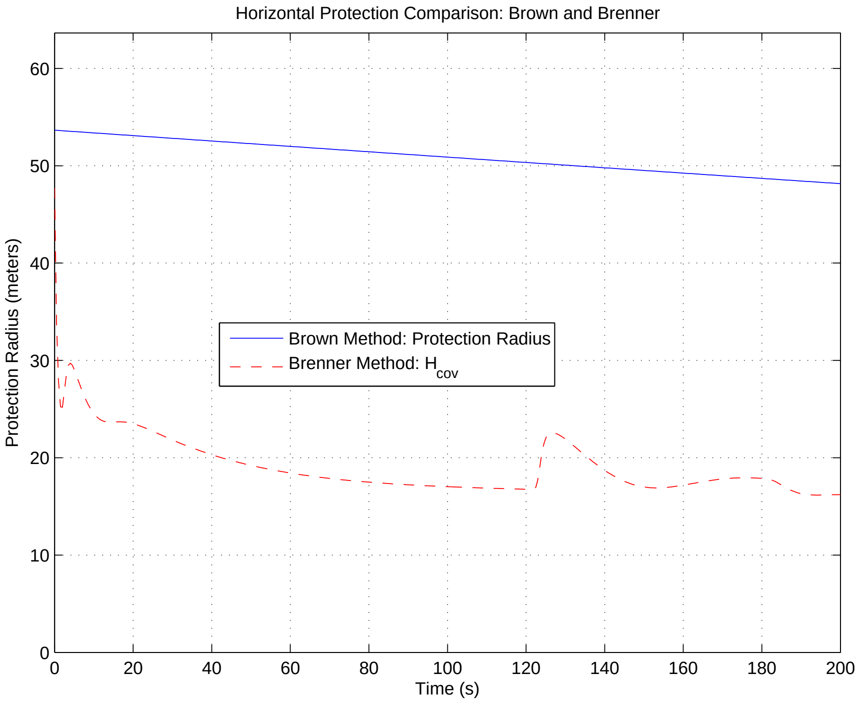

Figure 3.2: Comparison of HPL: RAIM and Solution Separation.

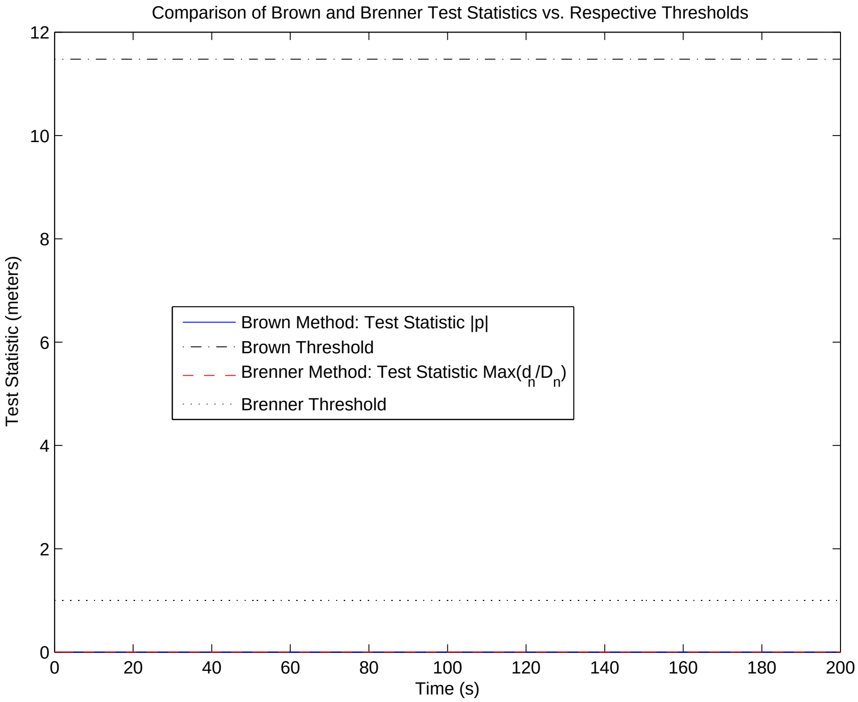

Figure 3.3: Comparison of Test Statistics: RAIM and Solution Separation.

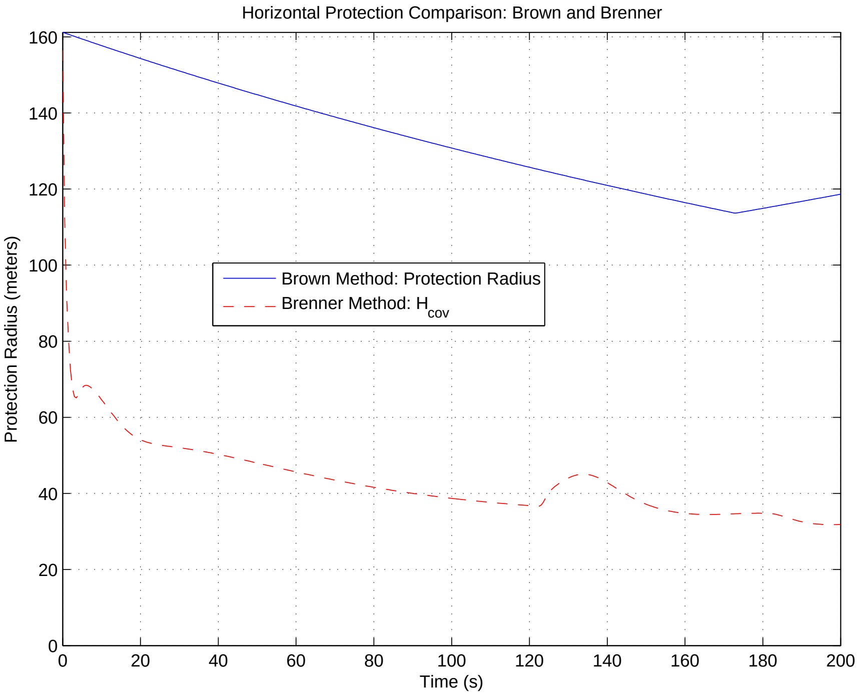

Figure 3.4: Comparison of HPL, 5 Satellites in View.

between 120 and 140 seconds, which coincides with the right-hand turning maneuver in the flight path. Even considering the dynamics, the solution separation algorithm benefits from using the Kalman filter produced covariance and not just the observation matrix in HPL determination. Although the overall HPL of the solution separation method has risen with the reduction in the number of satellites, it still satisfies the HAL requirement falling below 555.6 meters. No errors were induced or detected during this run, resulting in a test statistic evaluation mirroring that of Figure 3.3, and is thus not shown.

It is widely accepted [8; 11; 18; 30; 72] that snapshot algorithms, such as the aforementioned RAIM method, are capable of detecting step or jump bias like failures. Integrated systems are even more capable of detecting these failures, because the short-term stability of the INS eases identification of higher frequency disturbances in the measurements. As an illustration, Figure 3.5 shows the results of an intentional 100 m bias set to start at 60 seconds into the run, for duration of 50 seconds. With only five satellites

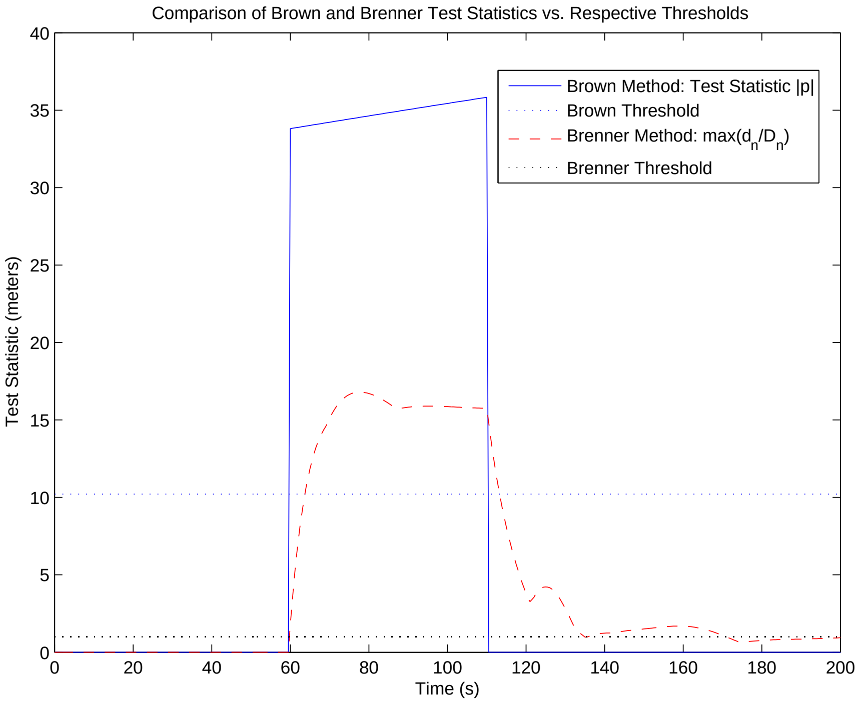

Figure 3.5: Bias Error, 5 Satellites in View.

available, both the RAIM algorithm and solution separation algorithm still easily detect the error.

It should be noted that the direction (i.e., sign) of the error impacts the ability of the integrity algorithm to detect any error, bias error included. The sign of the error impacts the computation of the difference between the predicted pseudorange and that of the pseudorange measurement, in that the polarity of the error combined with the polarity of the clock error may increase or decrease the pseudorange difference, influencing detection.

In contrast to bias-type errors, ramp-type errors are more difficult to detect [8; 11; 30; 72]. The nature of ramp errors growing slowly over time makes them more likely to appear as INS errors, and possibly be integrated by the Kalman filter into the position solution. Because of their potential impact on integrity, especially in precise navigation, they have been of particular interest to the navigation community in recent years [8; 11; 30; 31; 72; 129]. In general, faster ramp growth, greater than 5 m/s, is often sufficient for detection. Of particular concern are what as known as "slowly growing error" or SGE [8]. This project considered ramp errors of 0.5 m/s, 1 m/s, and 2 m/s. First comparisons were done between the RAIM method and the integrated system methods, then a comparison of the solution separation method and the initial implementation of the extrapolation method. For comparison between the RAIM and solution separation method, the results are presented for a 0.5 m/s ramp starting at 60 seconds into the run, for instances where there are seven, then five satellites available. The HPL for both cases are the same as were shown in Figure 3.2 and Figure 3.4 respectively, and therefore are not shown here.

Figure 3.6, the RAIM algorithm does detect the ramp error at approximately 120 s, or 1 minute after the error starts, with five satellite measurements available, and one in error. By comparison, the solution separation method detects the error at about 79 seconds, or 19 seconds after it began. Figure 3.7 is another test of the same ramp error, this time with seven satellites available. The results are only marginally better, with error detection occurring about $3-5$ seconds earlier for both methods. These results, and their impact on the phase of flight that can be met under the GPS integrity requirements for supplemental navigation point to the potential benefit of research in this area.

3.2.2 ***Comparison with Extrapolation Method****.*

Another method with potential in this area is the extrapolation method touched on earlier. Having examined briefly a fundamental RAIM algorithm and the solution separation method, which descends from it, the extrapolation method would at first appear to be a significant departure. However, like the solution separation method, the extrapolation method takes advantage of the inherent attributes of the Kalman filter in forming another promising integrity algorithm. In this study, the basic algorithm behind the extrapolation method has been implemented, but the multiple cycles averaging, to be explained next, is left to future work.

The extrapolation method focuses on the Kalman filter innovations as a source of information regarding the health of the incoming measurements. If the filter models are correct, and the measurements are acceptable, the innovations are expected to be zero

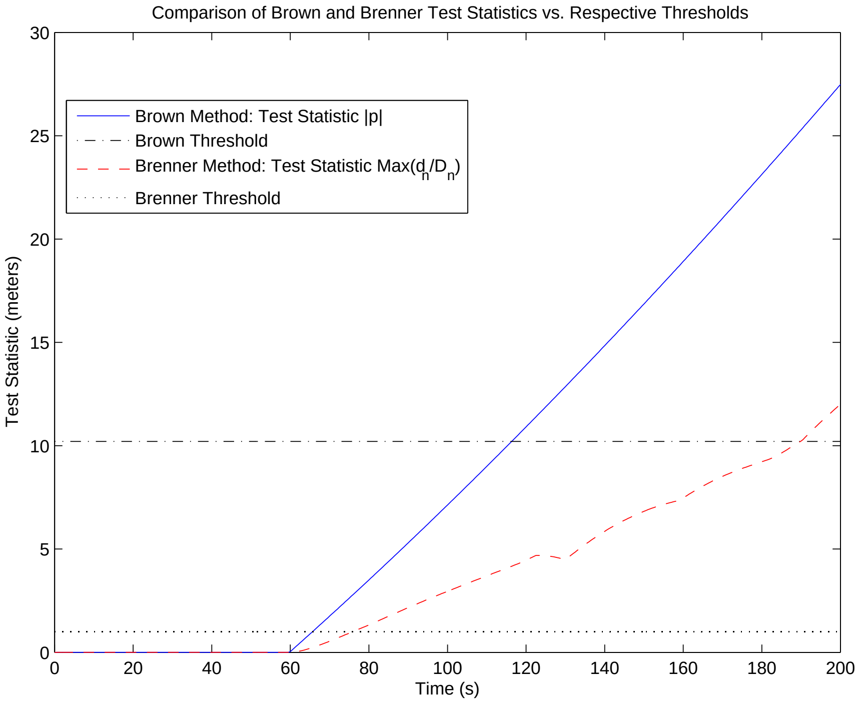

Figure 3.6: Ramp Error (0.5 m/s), 5 Satellites in View.

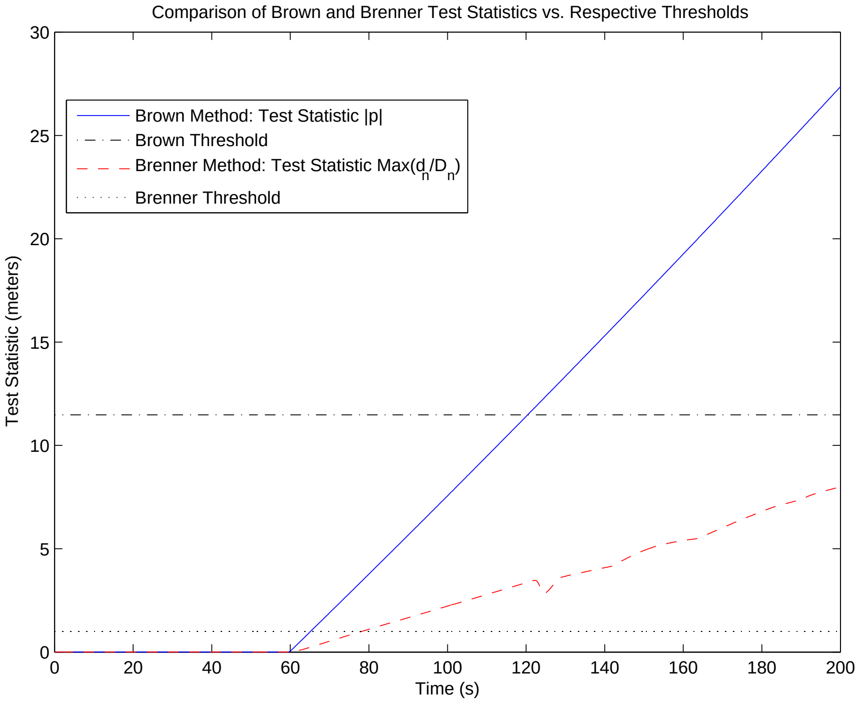

Figure 3.7: Ramp Error (0.5 m/s), 7 Satellites in View.

mean with a covariance V. Therefore, they provide a method of monitoring changes in the filter potentially indicative of errors in the measurements. Early descriptions of the extrapolation algorithm describe a bank of parallel filters, one for each satellite, all receiving the same measurements, but each set with large noise values for the satellite they were tracking [30]. More recent papers describe a filter bank like that of the solution separation method, with one filter processing all of the filter measurements, and parallel filters each excluding a single measurement. Filters are run in an open-loop, tightly coupled configuration using the standard Kalman filter equations.

The process involves averaging the measurements, which are the differences between the predicted pseudorange and the measured pseudorange, the observation matrix, and residuals over variable cycle times to reduce high frequency noise [31]. As mentioned earlier, the implementation of the extended cycle averaging is not complete as of this writing, but a basic implementation based on the integrity metrics that follow has been constructed to produce preliminary results. As in the case of RAIM, a test statistic is formed for comparison against a threshold to determine whether a failure is declared. In this case, the statistic is the quadratic formed by the innovations, $\mathbf{r}$, and the associated covariance matrix, $\mathbf{V}$ [31]:

$$\mathbf{V}(k) = \mathbf{H}(k)\mathbf{P}^{-}(k)\mathbf{H}^{T}(k) + \mathbf{R}(k) \tag{3.7}$$

$$\text{test statistic} = \mathbf{r}^{T}\mathbf{V}^{-1}\mathbf{r} \tag{3.8}$$

where k is the k^{th} cycle, $\mathbf{H}(k)$ is the observation matrix, $\mathbf{P}^{-}(k)$ is the propagated filter covariance, $\mathbf{R}$ is the estimated measurement covariance matrix, and $\mathbf{r}$ is the vector of innovations.

Alternatively, the innovations can be transformed to a vector s, through an eigenvector transformation, using the eigenvalues and modal matrix formed from $\mathbf{V}$. However, it can be shown that the test statistics are equivalent [31]:

$$\mathbf{s}^2 = \mathbf{s}^T\mathbf{s} = \mathbf{r}^T\mathbf{V}^{-1}\mathbf{r} \tag{3.9}$$

The fundamental HPL described in [31] parallels the basic RAIM method, where the ratio of the horizontal position error and the test statistic form a characteristic slope. In this case the horizontal position error is a function of the Kalman gain matrix, K, and the test statistic dividing it is formed from the eigenvector transformation matrices [31]:

Other literature [8; 72] has denoted this HPL term as HPL3, and identified it as only one of three terms used to define the HPL term in the extrapolation method. The additional HPL terms, and composite HPL are defined as [8; 72]:

$$\text{HPL1} = \sigma Q^{-1}(p_{fa}/(2n)) \tag{3.10}$$

$$\text{HPL2} = \sqrt{(\Delta x_0(1) - \Delta x_n(1))^2 + (\Delta x_0(2) - \Delta x_n(2))^2} \tag{3.11}$$

$$\text{HPL} = \text{Root sum square}(\max(HPL1, HPL2), HPL3) \tag{3.12}$$

where σ is formed from the elements related to horizontal position error in covariance matrix. The HPL2 term is actually the test statistic from the solution separation method and functions to drive the HPL upward when the growth between the filter horizontal position errors. More recent analysis by [8] suggests omitting this term on the basis that it fluctuates with the measurements and may not provide enough assurance in continuity of function. The HPL2 term may also cause violation the p_{fa} under certain conditions. Simulations did bear this out, with the overall HPL term tracking the ramp error and exceeding specifications. As a result, the implementation of the HPL used in this simulation is redefined as:

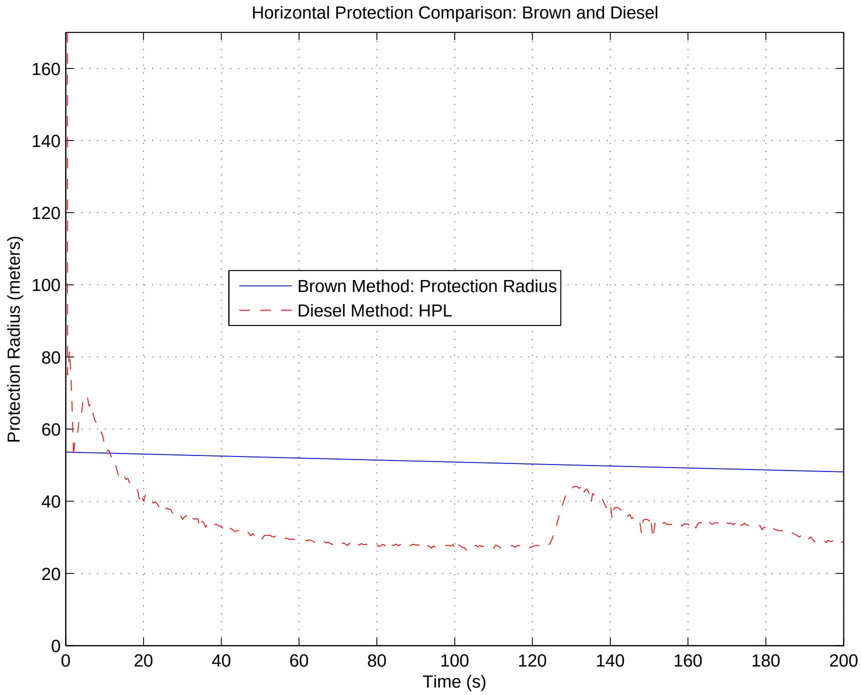

Figure 3.8: Comparison of HPL, 7 Satellites in View.

$$\text{HPL} = \sqrt{HPL1^2 + HPL3^2} \tag{3.13}$$

A relative comparison was done between a basic, non-averaging, extrapolation algorithm with the RAIM algorithm, as was done with solution separation earlier. The same numbers of satellites and types of ramp errors were used to provide a direct comparison. Figure 3.8 shows the HPL computed by each algorithm for the case of seven satellites in view. The HPL for the RAIM method is consistent with that seen in Figure 3.2. The HPL for the extrapolation method is clearly much lower than the RAIM method after the initialization of the filter covariance. This simulation performed as expected, yielding a better overall HPL than either the RAIM or solution separation methods based on [8; 72].

Figure 3.9 shows that the reduction in redundant measurements, from 7 to 5, doubles the HPL for both algorithms. There is no improvement seen in the detection time in Figure 3.10, but no degradation is apparent either, meaning the capability has not visibly diminished with the reduction in satellites even the uncertainty has grown in the protection

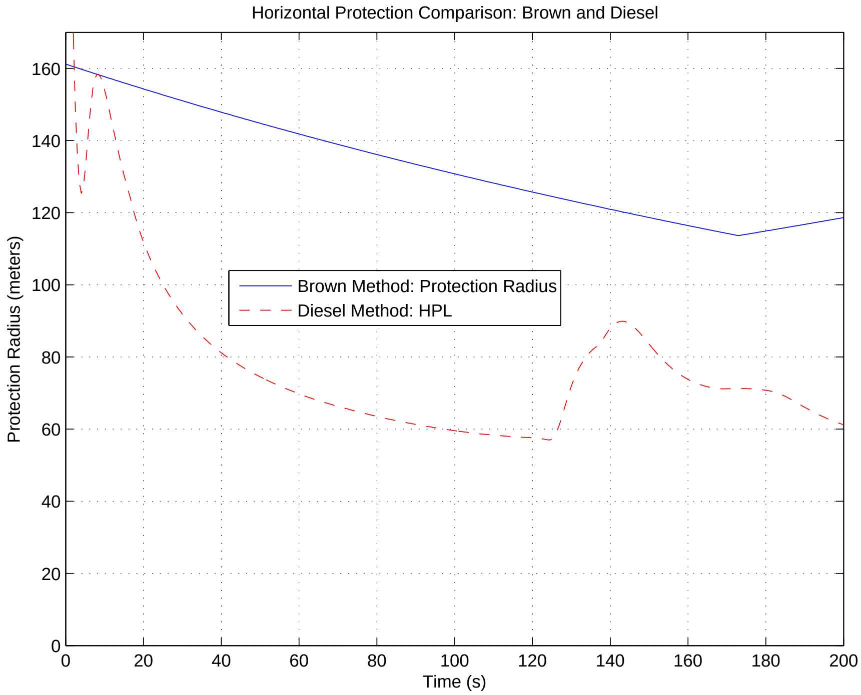

Figure 3.9: Comparison of HPL, 5 Satellites in View.

limit. Figure 3.11 provides a direct comparison between the solution separation method and the non-averaged extrapolation method, showing again that in this implementation, the solution separation method is detecting pseudorange error more quickly.

3.3 Conclusions

Figure 3.11 shows the time sequence of when the pseudorange errors drive the test statistic to the threshold for both the solution separation and extrapolation methods in this particular implementation. The differences between the two methods is more pronounced as the rate of ramp decreases. This is the fundamental reason behind the requirement to prefilter in the extrapolation method, and why work should continue on this model. These simulations verified assessments made by [6] that there may be probable cause to exclude the HPL2 term from the HPL formulation. Also, in the absence of averaging, the extrapolation method showed a high sensitivity to the estimated measurement noise was observed.

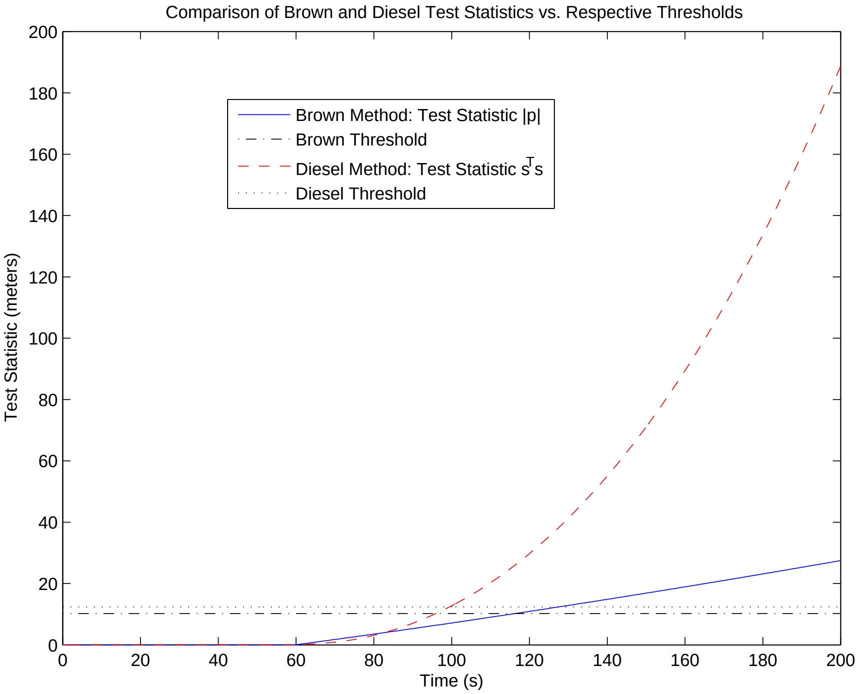

Figure 3.10: Ramp Error (0.5 m/s), 5 Satellites in View.

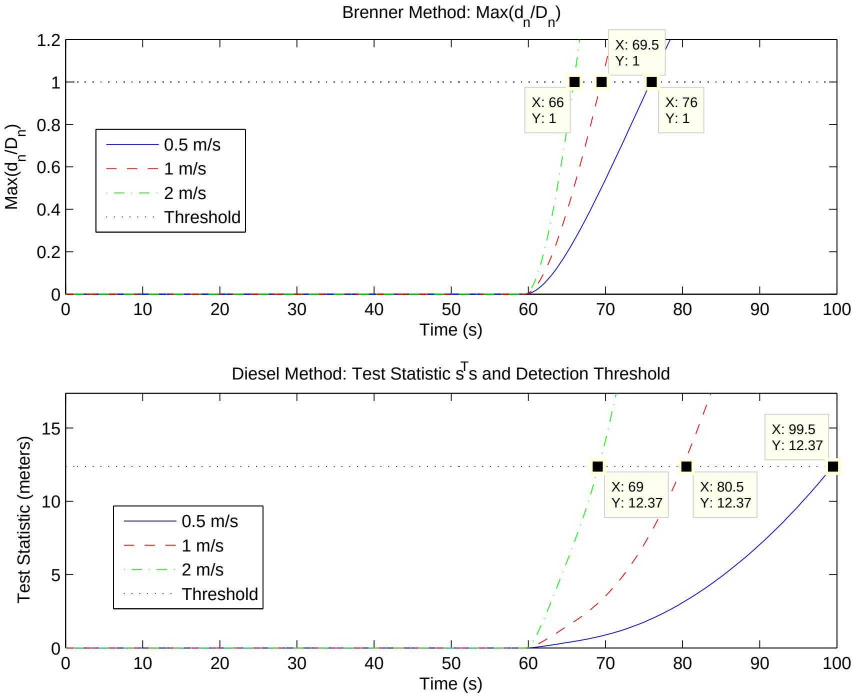

Figure 3.11: Comparison: SS vs. Extrapolation Methods, 5 Satellites in View.

These simulations provided insight into the critical part the estimate of the measurement noise plays in developing the innovation covariance, which feeds into both the test statistic and the filter gain, but not, perhaps, in equal degrees. The contribution to both obviously effects the slope calculation that aids HPL. Also, it can be shown that overly conservative estimates can greatly slow or even prevent detection because it artificially drives down the test statistic. A similar impact was observed for the basic RAIM case, as the estimated covariance impacts the threshold, as discussed earlier.

This chapter presented additional mathematical detail and described simulation development and results for the solution separation and extrapolation methods related to GPS integrity described in Chapter II. This examination of two state-of-the-art approaches to GPS integrity develops a deeper understanding of the GPS integrity process and provides a baseline for analysis in following chapters. The next chapter details the development of a new algorithm for image-based integrity.

IV. Image-Based Integrity Algorithm Development

This chapter presents a rigorously developed algorithm for estimating the potential worst case position error in an image-based navigation system due to an unanticipated bias influencing the pixel measurement in at least one or, quite possibly, both of the paired elements (e.g., x and y) in a pixel measurement, relative to an observable test statistic.

4.1 ***Motivation***

GPS integrity bases an inference of accuracy in the position solution on the quality of the GPS measurements [25]. While this would also be the case for image-aided navigation, user estimates of the error sources associated with GPS are well documented and readily accessible. This is not the case for the much newer image-aided technology. Also, the elemental measurement in GPS is the pseudorange to each satellite. Thus, each measurement is provided by a single source, the satellite, and can be discriminated from other measurements at each time epoch by the transmitted satellite code. However, in the case of image-aiding, each measurement source is a target translated to an image plane, and is presented as a measurement in the form of a vector containing a pair of pixel locations.

The pixel locations can be translated as two bearing measurements vice the single range measurement provided by GPS, for each source at each epoch. RAIM focuses on single fault conditions due to the complexity of identifying and excluding multiple simultaneous faults, although current research is focusing on the multiple failure case. This dissertation investigates the extension of the RAIM process to image-aided navigation. For clarity, discussions in this and following chapters will sometimes describe the image-based measurement equivalently as a pixel pair or target, with the latter reflecting the fact that the representation on the image plane is a projection of an observed stationary target. Consequently, in the image-based case the variable m refers to the number of pixel pairs available, or $2m$ total measurements.

4.2 Problem Set-up

4.2.1 ***Image Aiding****.* The notation used in this section is based on the integrated INS/Image-aided navigation scheme described earlier in Section 2.9 and Section 2.11. The focus in this research is the development of an image-based integrity approach in image-aided INS. With this in mind, the emphasis is on the image sensor measurements (in the form of target pixel coordinates). The fundamental concept proposes an epoch-by-epoch least-squares methodology independent of Kalman filter integration to monitor performance after an image feature tracking and correspondence algorithm has run. If the image-based measurements pass an epoch-by-epoch evaluation, they could then be fed into an INS/Image-aided system, but this was not done in this dissertation.

The concept proposes a least-squares methodology that would be external to the Kalman filter integration proposed [119] to monitor performance after an image feature tracking and correspondence algorithm has run. Algorithms such as Random Sample Consensus (RANSAC) [40], Maximum Likelihood Estimation Sample Consensus (MLE-SAC) [114], and stochastic projection [119], for example, already exist to aid in outlier rejection in feature matching and correspondence. However robust the algorithm, there remains the potential for measurement error to influence the navigation system.

This research is particularly focused on the potential for a biased pixel measurement being introduced into the navigation solution process after the initial processing, possibly as a result of image irregularities or a mistracked target. The concept of integrity in image-based navigation explored in this research requires that the potential impact of the error be quantified in terms of position error. This section illustrates the correlation between the real world target and its representation as a sensor measurement. Since the primary concern is on the measurement provided by the imaging system and not the imaging process itself, the following assumptions are made:

1. The coordinates of the stationary targets being tracked are known relative to an earth-fixed reference frame

2. A feature tracking algorithm is available to recognize common fixed objects from multiple images with known probability

3. The feature tracker provides measurements in the form of a pair of pixel coordinates for a tracked target with the measurements provided at a suitable frame rate

4. The camera is calibrated allowing a precise relationship between pixel coordinates and the corresponding position in the camera frame, with necessary corrections for lens distortion

5. The relationship between the camera mounting and vehicle body frame axes is known *a priori* and can be expressed as a direction cosine matrix, C_c^b

The imaging process is accomplished using a pinhole camera model, which aids in geometric and mathematical modeling by focusing all rays passing through the lens on a point called the perspective center [51]. In this model, the perspective center is used as the origin for the camera frame, with an imaginary image plane located one focal length in front to avoid inversion of the scene in the field of view of the camera.

The relationship between an object on the image plane and its actual real-world position, both relative to the camera frame origin, is that of similar triangle, making the mathematics tractable. The geometry of the example image-based navigation system is shown in Figure 4.1, using the parameters of interest described earlier in Table 2.1 and repeated here in Table 4.1, for clarity.

The mathematical description of these relationships can then be described by the following:

$$\mathbf{t}_m^n = \mathbf{s}_m^n + \mathbf{C}_b^n \mathbf{d}^b + \mathbf{p}^n \tag{4.1}$$

and as a result:

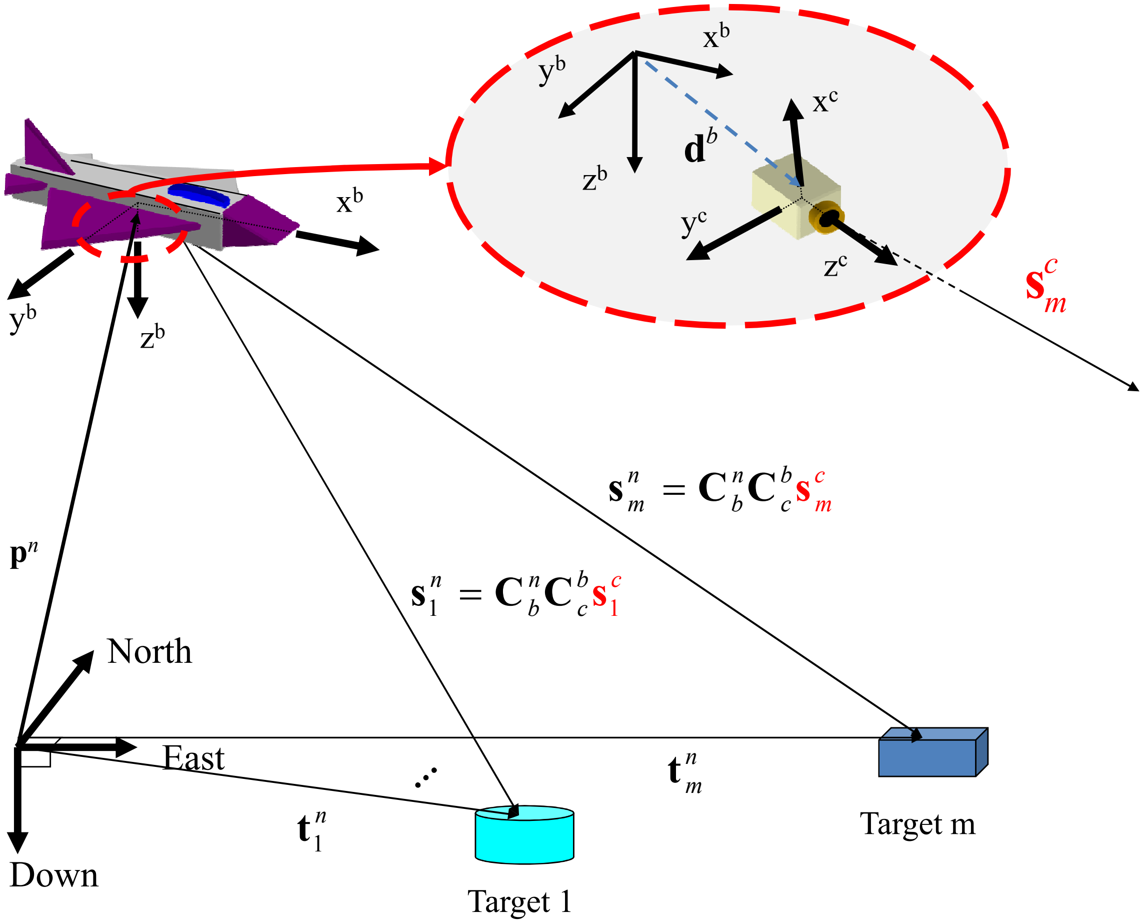

Figure 4.1: Vector Geometry in the Navigation Reference Frame.

Table 4.1: Parameter Definitions

Parameter	Description
$\mathbf{p}^n$	Vehicle position in navigation frame (NED)
$\mathbf{C}_b^n$	Vehicle body-to-navigation frame DCM
$\mathbf{s}_m^n$	LOS vector from the camera to m^{th} target (vector for each landmark currently tracked)
$\mathbf{t}_m^n$	Location of the m^{th} target (vector for each landmark currently tracked)
$\mathbf{d}^b$	IMU-to-Camera leverarm offset
$\mathbf{C}_c^b$	Camera-to-body frame DCM

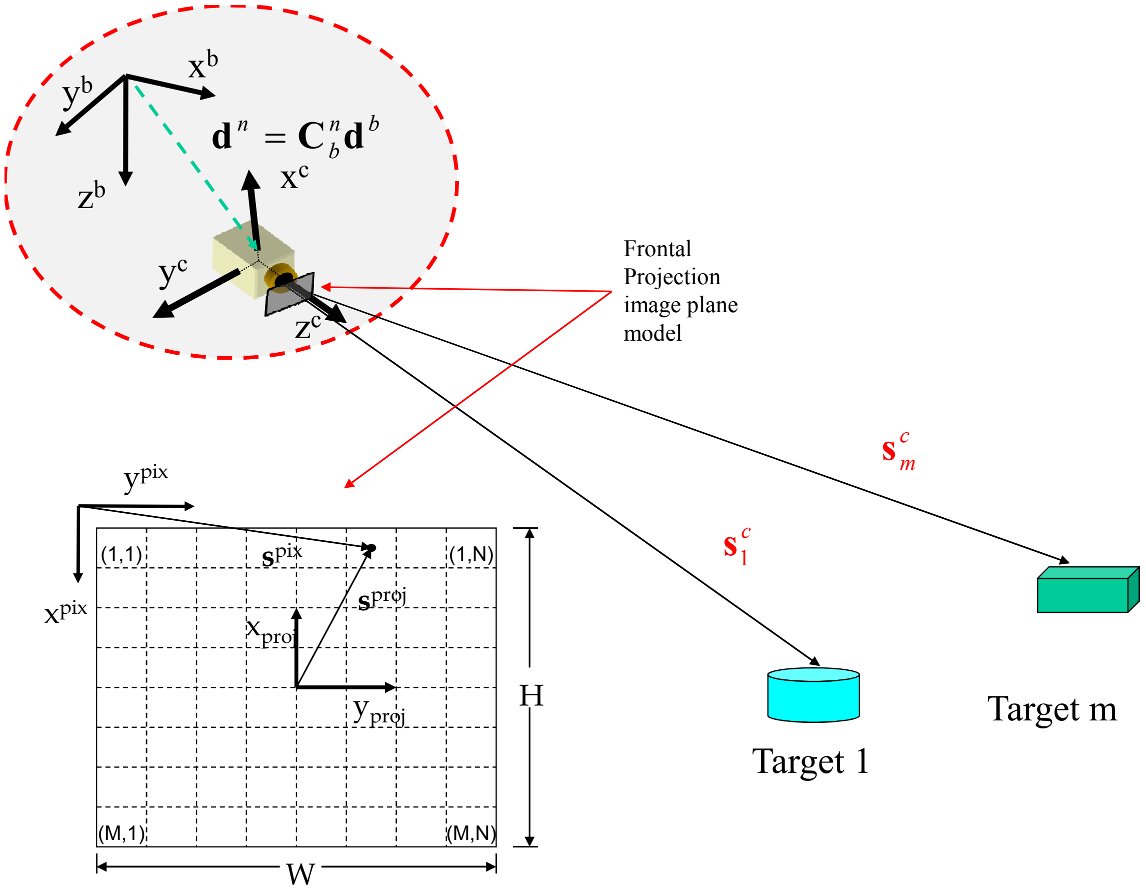

Figure 4.2: Transformation Between Camera Frame and Image Plane.

$$\begin{aligned}\mathbf{s}^n_{\mathbf{pt}_m} &\triangleq \mathbf{t}^n_m - \mathbf{p}^n \\ &= \mathbf{s}^n_m + \mathbf{C}^n_b \mathbf{d}^b \\ \mathbf{s}^n_{\mathbf{pt}_m} &= \mathbf{C}^n_b \left(\mathbf{C}^b_c \mathbf{s}^c_m + \mathbf{d}^b \right) \end{aligned} \tag{4.2}$$

The focus in the investigation of integrity is in the characteristics of the image-based measurements. Thus only the relevant details related to the measurement models are presented here (refer to Section 2.11 for additional details based on [121]). It is assumed that the locations of the items being tracked are known, which is a requirement for the single-epoch position solution that is considered in this research. If this is not the case, then a simultaneous localization and mapping algorithm like that proposed in [121] would need to be used to estimate these targets.

For the case where the targets are known, and accelerometer and gyro biases are assumed negligible, the error state vector given by Equation (2.102) can be reduced to:

$$\delta\mathbf{x} = \begin{bmatrix} \delta\mathbf{p}^n \\ \boldsymbol{\psi} \end{bmatrix} \tag{4.3}$$

where $\boldsymbol{\psi}$ are the tilt error states. The measurement model for the m^{th} feature is given by:

$$\mathbf{z}_m(t_i) = \mathbf{T}^{pix}_c \underline{\mathbf{s}}^c_m(t_i) + \mathbf{v}(t_i) \tag{4.4}$$

where $\underline{\mathbf{s}}^c_m(t_i)$ is the homogeneous form of the line-of-sight vector from the camera to target m, $\mathbf{T}^{pix}_c$ is the transformation matrix from the camera frame to the image frame, and $\mathbf{v}(t_i)$ is independent additive white Gaussian noise of zero-mean and covariance $\mathbf{R}$. Based on the assumptions stated for Equation (4.3), the Jacobian of the nonlinear measurement function, $\mathbf{h}[\mathbf{x}(t_i)] = \mathbf{T}^{pix}_c \underline{\mathbf{s}}^c_m(t_i)$, is reduced from that given by Equation (2.107) to the following linearized observation matrix:

$$\mathbf{H} = \begin{bmatrix} \frac{\partial \mathbf{h}}{\partial \mathbf{p}^n} & \frac{\partial \mathbf{h}}{\partial \boldsymbol{\psi}} \end{bmatrix} \tag{4.5}$$

where the elements of $\mathbf{H}$ are the partial derivatives given by the following equations based on [119], with $\mu = \frac{1}{s^c_{m_z}}$ and $\boldsymbol{\beta} = [0 \;\; 0 \;\; 1]$:

$$\frac{\partial \mathbf{h}}{\partial \mathbf{p}^n} = \mu \mathbf{T}^{pix}_c \left(\underline{\mathbf{s}}^c_m \boldsymbol{\beta} \mathbf{C}^c_b \mathbf{C}^b_n - \mathbf{C}^c_b \mathbf{C}^b_n \right) \tag{4.6}$$

and

$$\frac{\partial \mathbf{h}}{\partial \boldsymbol{\psi}} = \mu \mathbf{T}^{pix}_c \left(\frac{\partial \mathbf{s}^c_m}{\partial \boldsymbol{\psi}} - \underline{\mathbf{s}}^c_m \boldsymbol{\beta} \frac{\partial \mathbf{s}^c_m}{\partial \boldsymbol{\psi}} \right) \tag{4.7}$$

where

$$\frac{\partial \mathbf{s}^c_m}{\partial \boldsymbol{\psi}} = -\mathbf{C}^c_b \mathbf{C}^b_n \left[(\mathbf{t}^n - \mathbf{p}^n) \times \right] \tag{4.8}$$

For the integrity problem, the significance of this formulation is in the structure of the observation matrix compared to the GPS case. In the GPS case, information is known about the location of the satellite relative to the vehicle, which allows the measurement to be modeled as a pseudorange resulting in one $\mathbf{H}$ matrix row for each measurement. However, in the case of image aiding, the measurements are based on pixel pairs, resulting in multiple rows per target in the observation matrix (2-D on the pixel plane or 3-D homogenous in a camera frame). This structural change increases variability in the horizontal position, particularly when more than one pixel element is affected by error.

4.2.2 ***Revisiting RAIM.*** Fundamental terms associated with RAIM were introduced in Section 2.13.1. In developing the image-based integrity algorithm, some of the same mathematical definitions are chosen, including the parity matrix, $\mathbf{P}$ and the parity vector, $\mathbf{p}$, where the parity vector is derived using least-squares analysis and is used to evaluate the error using a projection onto the null-space of $\mathbf{H}^T$. Recall that RAIM often uses the term “slope” to describe the ratio between the horizontal position error and the calculated test statistic.

This RAIM slope is a ramp-like model error trajectory that linearly approximates relationship between a growing bias value and the impact of that bias on the horizontal position error. The goal of this research is to extend the RAIM concept to image-based integrity, so reiterate the baseline, the result of the RAIM slope formulation described in Section 2.13.1 is once again presented here. Under the single measurement failure case assumption in basic RAIM, the slope for the i^{th} element in the measurement vector can then be simply given as the mapping function between the horizontal position space and the parity space [130]:

$$Slope_i = \frac{\sqrt{\overline{\mathbf{H}}^2_{1,i} + \overline{\mathbf{H}}^2_{2,i}}}{\sqrt{\mathbf{S}_{ii}}} \tag{4.9}$$

where the subscripts 1 and 2 are associated with the north and east error directions, respectively, and S denotes the Parity matrix transpose times itself, resulting from the inner product of the parity vector. Since the worst case bias condition is assumed, an initial upper bound on the position error is estimated by using the measurement element associated with the largest slope and is given by [130]:

$$|H_{bias}| = \max_{i=1:m} \left[Slope_i \, \right] \|\mathbf{p}\| \tag{4.10}$$

where $\max\left[Slope_i \, \right]$, $i = 1{:}m = 1, 2, ..., m$ is alternatively called $SLOPE_{\max}$. Of course, the measurements are not deterministic, and thus the process cannot reflect the true horizontal position error. However, statistically, if the measurement noise is assumed to be additive white Gaussian noise (AWGN), then the expected value of the parity vector is reasonably approximated by the deterministic case [110; 117]. Ultimately, the slope allows for a visualization of the projection of error into the observable parity space based test statistic in the form of a marginal density. It should also be noted that original bias magnitude term is divided out in the final ratio.

4.3 Image-Based Integrity Algorithm Development

The goal of this research is to extend RAIM-like methodologies and algorithms (described in Section 2.13.1) to the image-based navigation problem. Simplifying assumptions are made to allow direct comparison of an image-based approach with a GPS RAIM-type approach, which specifically focuses on the position states. In understanding an approach to image-based navigation integrity, it is important to realize that one of the fundamental elements given in RAIM cannot be pre-determined in image-aiding. That element is geometry prediction. With this in mind, the algorithm being developed here takes into consideration the nature of the image-based measurement as an angular type measurement and incorporates this knowledge into the algorithm. The following simplifying assumptions are noted:

- Tracked target positions are known *a priori* and do not need to be estimated in the state vector.
- A image-based measurement is now considered a two element set (taking advantage of association of x/y pairs) where potentially both elements have failed
- Bias is multi-dimensional and assumed to be a bias magnitude times sinusoidal components of error angle relative to the true pixel location
- Noise is assumed to be zero-mean additive white Gaussian noise when present, but the investigation primarily examines the deterministic case to parallel GPS RAIM Slope approach and form a fundamental baseline

The single failure assumption described earlier in RAIM is not generally valid for the image-based case. In the image-based case, if there is an error in target correspondence (i.e., the wrong target is identified), then this error will generally affect both the x and y

pixel coordinates. Therefore, a multi-dimensional evaluation space is required in order to detect pixel pair failures.

One key advantage in the image-aided case is that an x and y elements of a pixel pair are linked to a single observed target and, therefore, hold adjacent positions, i and j, within the measurement vector. The bias components b_i, b_j are modeled as polar components of the bias vector $\mathbf{b}$, with $b_i = ||\mathbf{b}|| \sin(\theta)$ and $b_j = ||\mathbf{b}|| \cos(\theta)$. The choice of $\sin$ or $\cos$ was made with consideration of the definition of x and y directions in the image plane.

The concept of "Slope" described by [18] can be fundamentally restated as the ratio of the squared vector norm of the horizontal position error to the squared vector norm of the parity vector, or equivalently, the residual vector in a deterministic relationship:

$$\frac{\|\delta \mathbf{x}_h\|^2}{\|\mathbf{p}\|^2} = \frac{\delta \mathbf{x}_h^T \delta \mathbf{x}_h}{\mathbf{p}^T \mathbf{p}} = \frac{\mathbf{b}^T \overline{\mathbf{H}}_h^T \overline{\mathbf{H}}_h \mathbf{b}}{\mathbf{b}^T \mathbf{P}^T \mathbf{P} \mathbf{b}} \tag{4.11}$$

where $\overline{\mathbf{H}}$ is the $n \times m$ pseudoinverse matrix of $\mathbf{H}$ and $\mathbf{P}$ is the parity matrix, both described in Section 2.13.1. The subscript h has been used to indicate the horizontal position elements, including the first two rows of the $\delta \mathbf{x}$ column vector and corresponding rows of $\overline{\mathbf{H}}$. The slope formulation is often visualized in the square root of the ratio, as it is a monotonic function. The following analysis is accomplished before taking any square root. Letting $\mathbf{G} = \overline{\mathbf{H}}_h^T \overline{\mathbf{H}}_h$ and $\mathbf{S} = \mathbf{P}^T \mathbf{P}$ Equation (4.11) can be further simplified:

$$\frac{\|\delta \mathbf{x}_h\|^2}{\|\mathbf{p}\|^2} = \frac{\mathbf{b}^T \mathbf{G} \mathbf{b}}{\mathbf{b}^T \mathbf{S} \mathbf{b}} \tag{4.12}$$

Following the assumption that the bias vector is all zeros except for the b_i and b_j components, the numerator of the fraction reduces to:

$$\mathbf{b}^T \mathbf{G} \mathbf{b} = b_i^2 G_{ii} + b_i b_j \left(G_{ij} + G_{ji}\right) + b_j^2 G_{jj} \tag{4.13}$$

The $\mathbf{G}$ matrix is symmetric, thus the term $(G_{ij} + G_{ji}) = 2G_{ij} = 2G_{ji}$. Using this knowledge, rearranging terms and replacing the b_i, b_j components with their sinusoidal definitions yields:

$$\begin{aligned} \mathbf{b}^T\mathbf{G}\mathbf{b} &= \|\mathbf{b}\|^2 \sin^2(\theta) G_{ii} + \|\mathbf{b}\|^2 \cos^2(\theta) G_{jj} \\ &\quad + \|\mathbf{b}\|^2 \sin(\theta)\cos(\theta)(2G_{ij}) \end{aligned} \tag{4.14}$$

The trigonometric identity for the product of the sin and cos, $\sin A \cos B$, is computed as $\frac{1}{2}\left[\sin(A+B) + \sin(A-B)\right]$, with $A = B = \theta$ is then used to produce a final form of the numerator, related in the first form:

$$\|\delta\mathbf{x}_h\|^2 = \|\mathbf{b}\|^2 \left[\sin^2(\theta)G_{ii} + \sin(2\theta)G_{ij} + \cos^2(\theta)G_{jj}\right] \tag{4.15}$$

The denominator is derived in a similar manner, noting that the $\mathbf{S}$ matrix is also symmetric, resulting in the following:

$$\|\mathbf{p}\|^2 = \|\mathbf{b}\|^2 \left[\sin^2(\theta)S_{ii} + \sin(2\theta)S_{ij} + \cos^2(\theta)S_{jj}\right] \tag{4.16}$$

Dividing the numerator by the denominator negates the bias term, leaving the ratio of the horizontal position magnitude to the parity vector magnitude to be described by:

$$\frac{\|\delta\mathbf{x}_h\|}{\|\mathbf{p}\|} = \left[\frac{\sin^2(\theta)G_{ii} + \sin(2\theta)G_{ij} + \cos^2(\theta)G_{jj}}{\sin^2(\theta)S_{ii} + \sin(2\theta)S_{ij} + \cos^2(\theta)S_{jj}}\right]^{\frac{1}{2}} \tag{4.17}$$

This result redefines the classic "standard slope" relationship to include a function of the angular error relative to the expected pixel pair location in the image plane. This formulation was independently derived based on the angular error definitions assumed in this research, but later found to be mathematically similar to a result given in [53]. Unlike the previous representation, this new slope definition explicitly expresses the numerator and denominator in terms of θ, providing more insight into angular error influence. In addition, using the trigonometric identity $\cos^2(\theta) = 1 - \sin^2(\theta)$, the ratio in Equation (4.17) can be rewritten in terms of $\sin$ only:

$$\frac{\|\delta\mathbf{x}_h\|}{\|\mathbf{p}\|} = \left[\frac{\sin^2(\theta)\,(G_{ii} - G_{jj}) + \sin(2\theta)G_{ij} + G_{jj}}{\sin^2(\theta)\,(S_{ii} - S_{jj}) + \sin(2\theta)S_{ij} + S_{jj}}\right]^{\frac{1}{2}} \tag{4.18}$$

Evaluating this form for $\theta = \pi/2$ and $\theta = 0$ yields the following interesting results:

$$\begin{aligned}\frac{\|\delta\mathbf{x}_h\|}{\|\mathbf{p}\|} &= \left[\frac{\sin^2(\frac{\pi}{2})\,(G_{ii} - G_{jj}) + \sin(\pi)G_{ij} + G_{jj}}{\sin^2(\frac{\pi}{2})\,(S_{ii} - S_{jj}) + \sin(\pi)S_{ij} + S_{jj}}\right]^{\frac{1}{2}} \\ &= \left[\frac{(1)\,(G_{ii} - G_{jj}) + (0)G_{ij} + G_{jj}}{(1)\,(S_{ii} - S_{jj}) + (0)S_{ij} + S_{jj}}\right]^{\frac{1}{2}} \\ &= \left[\frac{G_{ii}}{S_{ii}}\right]^{\frac{1}{2}}\end{aligned} \tag{4.19}$$

for $\theta = \pi/2$, and

$$\begin{aligned}\frac{\|\delta\mathbf{x}_h\|}{\|\mathbf{p}\|} &= \left[\frac{\sin^2(0)\,(G_{ii} - G_{jj}) + \sin(0)G_{ij} + G_{jj}}{\sin^2(0)\,(S_{ii} - S_{jj}) + \sin(0)S_{ij} + S_{jj}}\right]^{\frac{1}{2}} \\ &= \left[\frac{(0)\,(G_{ii} - G_{jj}) + (0)G_{ij} + G_{jj}}{(0)\,(S_{ii} - S_{jj}) + (0)S_{ij} + S_{jj}}\right]^{\frac{1}{2}} \\ &= \left[\frac{G_{jj}}{S_{jj}}\right]^{\frac{1}{2}}\end{aligned} \tag{4.20}$$

for $\theta = 0$. The choice of θ as either $\pi/2$ or 0 implies the cases where there is an angular error strictly in x or y directions defined in the pixel plane coordinates, relative to each expected element of the pixel pair. As a result, Equations (4.19) and (4.20) show that the "new slope" definition reduces to the standard slope for an error only on the i^{th} or j^{th} measurements, respectively.

This chapter developed a rigorous baseline image-based integrity algorithm for evaluating the potential impact of horizontal position error as a result of measurement bias on one or both elements of a pixel pair. The next chapter presents results and analysis of specific image-based measurement scenarios including cases where only position and both position and attitude are being estimated by the image-based navigation system.

V. *Results and Analysis of Image-Based Integrity Algorithm*

This chapter details scenarios used for evaluation and provides results and analysis based on the fundamental developments of Chapter IV.

5.1 *Simulation Development and Parameter Definition*

Although many potential applications exist for an image-based integrity method, this research is particularly focused on the integrity risk posed in an air-to-air refueling (AAR) scenario. With this in mind, two primary modes of operation are considered for evaluation: 1) Estimating position only, and 2) Estimating both position and attitude. In estimating position only, there is an assumption that a high-quality, navigation-grade Inertial Measurement Unit (IMU) with very accurate attitude determination is used, in which case the attitude errors are so small that they can be neglected, eliminating the need to estimate attitude and related bias errors for this investigation.

For example, when considering the air-to-air refueling scenario, it is believed that reasonable magnitudes of error could be achieved at the short distances (e.g., tens of meters) in question. As a rough approximation, consider $Position\ Error\ (m) \approx Angular\ error\ (rads) \times Range\ to\ Target\ (m)$, then for an error of 0.1 degree the resulting position error would be $0.00175 \times 15m = 0.026m$, or less than 3 centimeters, and would be expect to decrease in accordance with a decrease in distance. For comparison, if the angular error increased to 360 degrees, the resulting position error would grow to 94m at the same distance, clearly unacceptable. The choices of 0.1 degree and 360 degrees are examples chosen to simply illustrate coarse position error differences as a function of attitude error using extremes based on 1-sigma deg/hr gyro biases in low-end navigation grade (or high-end tactical grade) and high-end Micro-Electromechanical Systems (MEMS) IMUs, from survey data in [112].

To be clear, the term bias used throughout this chapter only refers to pixel error, representing the case where a given feature is misidentified, not to gyro or accelerometer bias.

When estimating both position and attitude, the error state vector and associated observation matrix are expanded to include elements associated with attitude errors, increasing the dimensionality of both $\delta\mathbf{x}$ and $\mathbf{H}$, respectively. This case still excludes gyro and accelerometer bias states. Assumptions previously given in Section 4.2.1 and Section 4.3 are still held, unless otherwise stated.

The image-aided equations provided in Section 2.11 are used in simulation to build sets of known target locations in a local reference frame, relative to a user defined vehicle position. The discrete targets are representative of stationary landmarks at known locations, and are observable within the field of view of a calibrated camera. The direction cosine matrix (DCM) between the camera and body is assumed fixed and known. The DCM between the body and navigation frame is developed without error based on the previous IMU quality assumption.

As shown in Section 4.3, a pixel bias is expressed in polar form and added to both the x and y components of different pixel pairs, influencing one pixel pair at a time. The pixel error is perceived as a bias on the pixel plane, potentially the result of a mismatched target in a discrete tracking problem. The polar expression seems well suited for this application in that an error pixel location can be reasonably described as laying at some angle θ on a circle of radius r from the true pixel location. The $\mathbf{H}$ matrix is initially fixed at the truth value to avoid non-linearities under repeated angular cycles of generated samples. A study of linearization effects is included later in this chapter.

In reality, a wide ranging number of target locations may be observable, depending on the scene in view. For the purposes of this research, between four and ten targets were used. The choice to use a small target set was made to more readily demonstrate the impact of measurement error in the image-based case, with the minimum four pixel pairs serving as the basis for “worst case” scenarios. Adding more image-based measurements (i.e., increasing redundancy) without adding additional errors further constrains the least squares problem, and potentially improve the navigation solutions, as will be discussed later. Therefore, using a reduced number of redundant equations allows for a more realistic examination of problematic scenarios, as well as providing clarity in the plotted results. The approaches described here can be used with any greater number of measurements.

Table 5.1 lists six example baseline configurations of four pixel pairs each that are used for comparisons throughout this chapter. Included in the table are three orthogonal configurations and three different asymmetric configurations, based on a 768×1024 pixel plane, which are illustrated in Figure 5.1. This list is not all inclusive of all possible pixel locations, as will be shown in later discussions of random target sets, but is designed to provide a simple reference for later analysis.

Table 5.1: Measurement Geometry Designation and Pixel Coordinates

Geometry Identifier	Geometry Description	Pixel Pair Coordinates
$1a$	Orthogonal & Symmetric	$(384.5, 256.25), (384.5, 768.75),$ $(192.25, 512.5), (576.75, 512.5)$
$1b$	Orthogonal & Symmetric/Equidistant	$(384.5, 312.5), (384.5, 712.5),$ $(184.5, 512.5), (584.5, 512.5)$
$1c$	Orthogonal & Symmetric/Equidistant/Rotated	$(525.92, 371.08), (243.08, 653.92),$ $(243.08, 371.08), (525.92, 653.92)$
$2a$	Non-Orthogonal & Asymmetric (Case 1)	$(559, 919), (200, 460),$ $(123, 725), (630, 212)$
$2b$	Non-Orthogonal & Asymmetric (Case 2)	$(708, 652), (658, 117),$ $(234, 781), (398, 650)$
$2c$	Non-Orthogonal & Asymmetric (Case 3)	$(271.88, 362.39), (271.88, 724.78),$ $(135.94, 362.39), (407.82, 362.39)$

Extensive simulations were run using various geometries, evaluating different bias levels and random angles drawn from a uniform random distribution of at least 2000 samples with $\theta \in [0, 2\pi)$. Selected examples are provided in the following sections. The results are shown for deterministic cases with the relational outcomes (e.g., pixel bias,

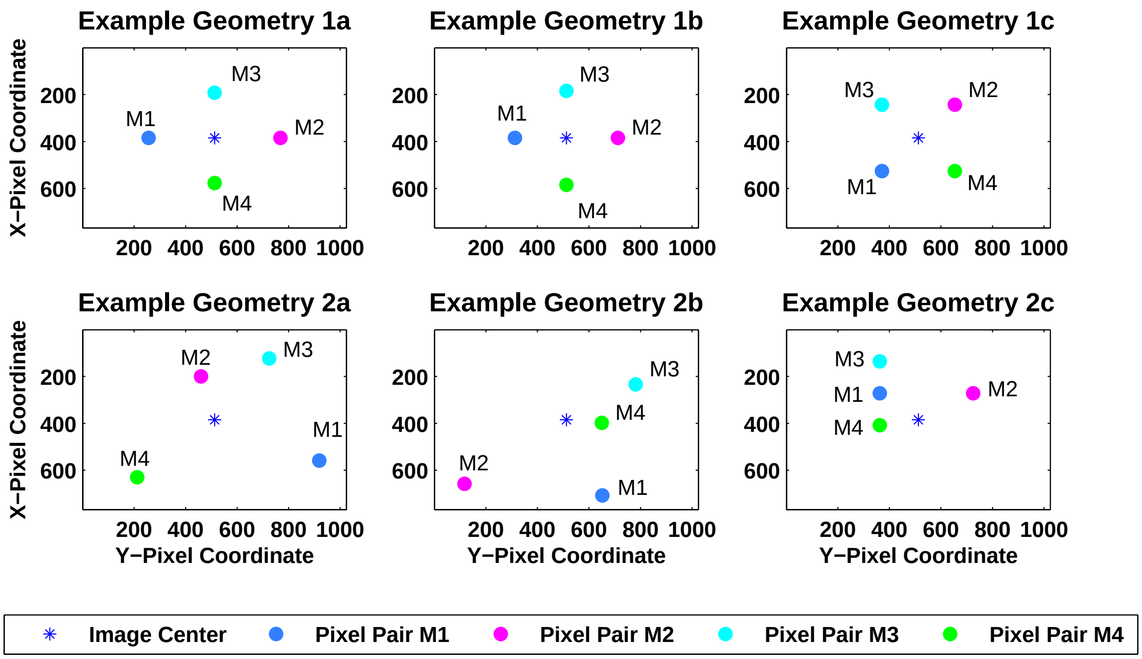

Figure 5.1: Illustration of Pixel Geometries for Table 5.1.

horizontal position error, parity vector magnitude) shown for all angles on each example plot. Ideally, the goal is to establish a slope trajectory that predicts the line of travel for the horizontal position error resulting from the worst case angular error. This line would relate to a minimal test statistic value, and thus be in the left or least direction side in a horizontal position error vs. test statistic representational plot. Comparative performance analysis is offered between the new algorithm and accepted GPS RAIM formulations under selected example geometries. For clarification, the following definitions of figure legend entries are provided, where m is the number of pixel pairs:

- **Max Slope**: The maximum single slope calculation based on standard RAIM (Equation (4.9))
- **Max-Max Slope**: The minimum value chosen between the two computations:
  1. The sum of the two largest single slopes based on standard RAIM, and

2. $Slope_{MAXMAX}$ computed using eigenvalue decomposition for two element combinations of possible bias locations taken two at a time (e.g., (b_1, b_2), (b_1, b_3), (b_1, b_4),. . .,(b_1, b_j),(b_2, b_3),(b_2, b_4),. . .,(b_2, b_j),(b_3, b_4),(b_3, b_5). . .,(b_i, b_j) for $i = 1, 2, \ldots, 2m - 1; j = i + 1, i + 2, \ldots, 2m; i \neq j$), based on [17]

- **Max**$\{i^{th}$ **+** $j^{th}\}$ **Slope**: The minimum value chosen between the two computations:

1. The sum of the individual standard RAIM slopes of the two adjacent single slopes associated with the non-zero bias terms (b_i, b_j) for $i = 1, 3, 5, \ldots, 2m-1; j = i + 1$, and

2. $Slope_{MAXMAX}$ computed using eigenvalue decomposition for adjacent two element combinations of possible bias locations (e.g., (b_1, b_2), (b_3, b_4), . . .,(b_i, b_j) for $i = 1, 3, 5, \ldots, 2m - 1; j = i + 1$), derived based on [17]

- **New Slope**: The maximum slope computed using Equation (4.18) and worst case θ value

5.2 *Estimating Position Only*

The investigation begins by examining performance of the “new slope” formula under the first mode of operation, estimating position only. As a result of the high-quality IMU assumption described in Section 5.1, Equations (4.3) and (4.5) are reduced to the following forms:

$$\delta \mathbf{x} = \left[\begin{array}{c} \delta \mathbf{p}^n \end{array} \right] = \begin{bmatrix} \delta \mathbf{x}_n \\ \delta \mathbf{x}_e \\ \delta \mathbf{x}_d \end{bmatrix} \tag{5.1}$$

and

$$\mathbf{H} = \left[\frac{\partial \mathbf{h}}{\partial \mathbf{p}^n}\right] \tag{5.2}$$

5.2.1 ***Example 1: Orthogonal Geometry***. The first simulation case examines four pixel pairs arranged orthogonally to each other as shown in Figure 5.2, where an error has been induced on the second pixel pair. The second pixel pair was chosen after evaluating Equation (4.18) over all values of θ from 0 to 2π radians in steps of $1/N$, where N is the number of samples, and taking the maximum value from the set ratios of all pixel pairs, which occurs at $\theta = 0$ on pair number two. Figure 5.3 shows this representation, with the solid sinusoidal line reflecting the computed values of Equation (4.18) and the the dashed lines reflecting the results of Equation (4.9). The sinusoid results from correlation in the x and y directions.

Recall that from Equations (4.19) and (4.20) that the standard slopes and new slopes generated for each pixel pair intersect at 0 and $\pi/2$ radians. This is clearly seen in Figure 5.3 for orthogonal configuration $1a$. In addition, due to the fact that the target representations on the pixel plane are aligned with the image plane cardinal directions, the standard slope values appear to bound the new slope.

As one might expect, the orthogonal positioning lends nearly equal weight to the influences of each pixel pair on the position determination. As a result, radial error of some bias magnitude about these locations presents a reasonably consistent level of horizontal position error. This phenomena is illustrated in Figure 5.4. In this figure, and all similar type representations, the position errors resulting from all angular errors are simultaneously plotted for a given bias level. The result is an quasi-elliptical pattern of deterministic position error values of each sample plotted against the test statistic values for the same sample. Under the orthogonality condition, the quasi-elliptical pattern is thinned to the point of almost appearing as a line as shown in Figure 5.4.

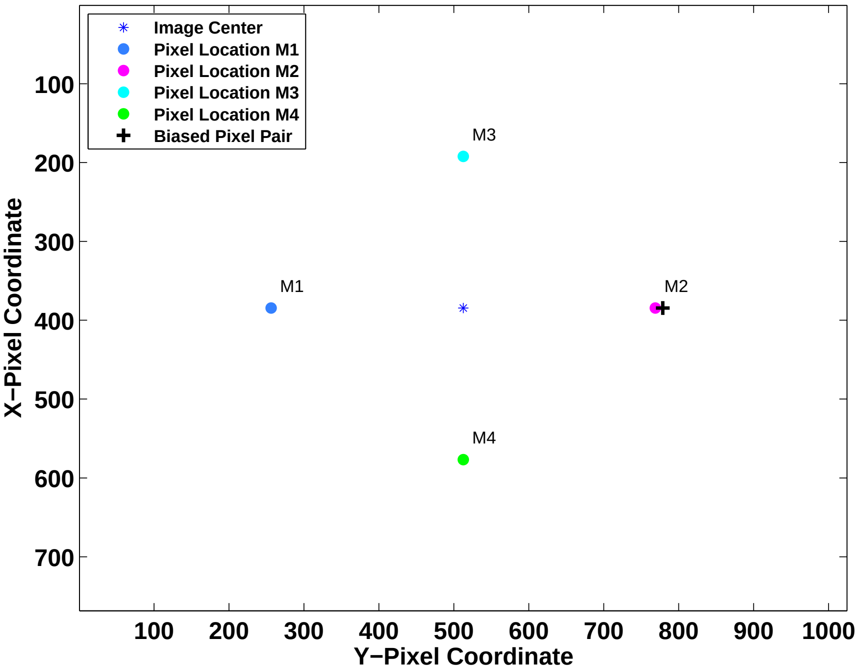

Figure 5.2: Pixel Measurement Geometry for Example 1 (Configuration $1a$).

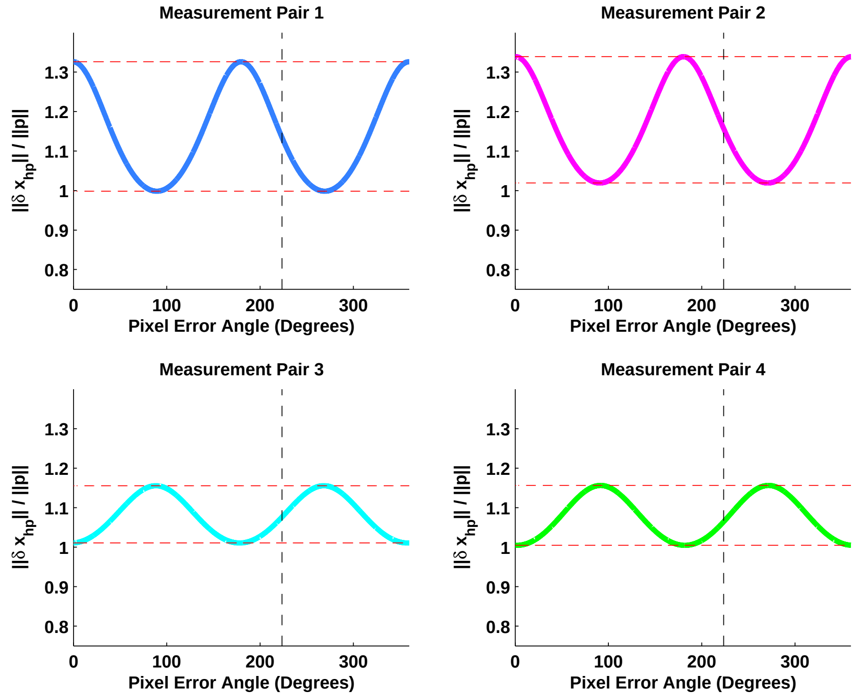

Figure 5.3: New and Standard Slope Comparison Over Varying Angles; Orthogonal Geometry ($1a$). NOTE: The "New Slope" in each subplot is indicated by the solid line, while the "Standard Slopes," which produce the same result as the "New Slope" evaluated at both $\theta = 0$ and $\theta = \pi/2$, are indicated by the dashed lines.

Slight variations in the both the orientation and length (relative to the test statistic axis) of the pattern stem from minute differences in distance from center of the pixel locations. The different slopes defined in the previous subsection are illustrated in Figure 5.4. It is clear to see that both the Max-Max Slope and Max$\{i^{th}$ + $j^{th}\}$ Slope are extremely conservative estimates of the potential error relationship. While the conservative estimate still bounds the potential error, it negatively impacts the potential availability. In contrast, the standard Max slope and new slope both lie tangent to the the leftmost position error. It should be noted that the Max-Max Slope and Max$\{i^{th}$ + $j^{th}\}$ Slope represented in this plot are the computations based on the sums of the single RAIM slopes, as they are the more conservative estimates for comparison. In fact, the $Slope_{_{MAXMAX}}$ based on [17] produces an undefined or “infinite” slope value in this scenario, and is thus not shown. This is due to the precise orthogonality of the pixel pairs combined with the pixel pair correlation, where the latter is lost through the process of taking two elements at a time that are not necessarily adjacent and may in fact be identical, causing a singularity in the computation.

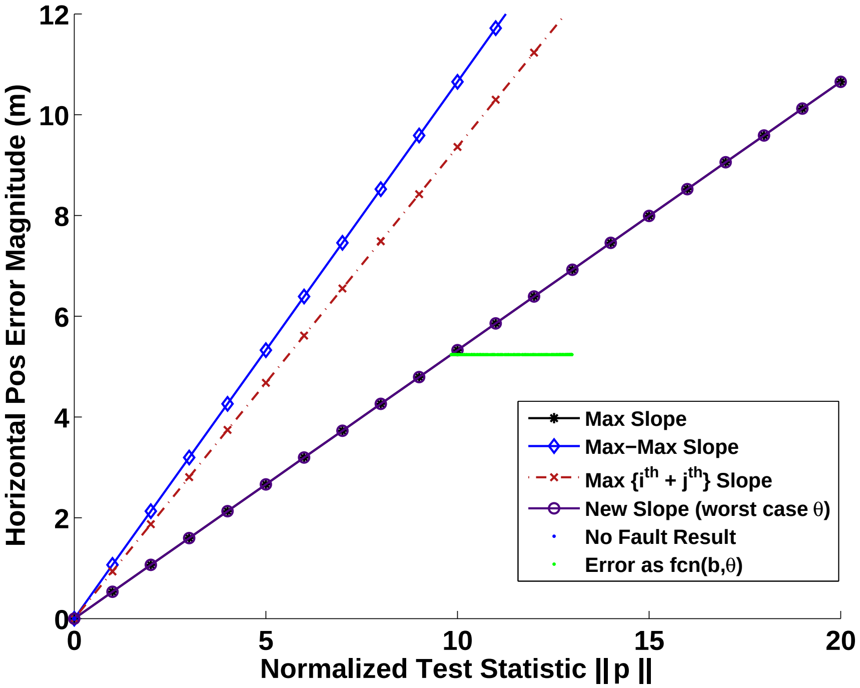

Figure 5.4: RAIM-like Presentation for Comparison; Orthogonal Geometry ($1a$). NOTE: "Max Slope" and "New Slope" overlay each other in this plot.

When pixel locations are rotated about the center of the pixel plane, the condition described where the standard slope appears to bound the new slope no longer holds, even though the pixel locations remain orthogonal. In these cases, the standard slope falls below the maximum of the new slope. However, the new slope remains to the leftmost side of the position error values while the standard slope falls below the new slope, which means the new slope approach does a better job of describing the worst-case error. This observation is shown in Figure 5.5 for the case where the pixel pair geometry has been rotated $\pi/4$ radians, as in configuration $1c$.

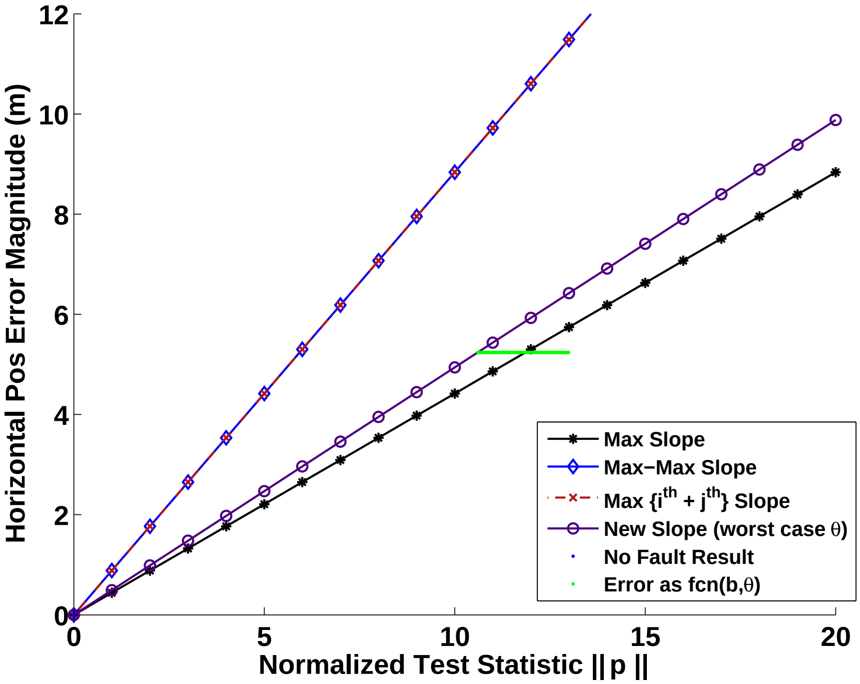

Figure 5.5: RAIM-like Presentation for Comparison; Rotated Pixel Pairs (Configuration $1c$).

5.2.2 ***Example 2: Asymmetric Geometry***. Limitless possible target geometries exist, but an example is chosen here to represent a reasonable condition where the four pixel pairs are located roughly in four quadrants of the image plane. Although no longer orthogonal or symmetric in distribution, the configuration would seem reasonable for admission into a potential position solution based on geometric criteria. The geometry is illustrated in Figure 5.6, with the new slope computation shown in Figure 5.7. The maximum new slope value corresponds to the peak value in Figure 5.7 for pixel pair four at the $\frac{4\pi}{5}$ angular value. Note that the scale in Figure 5.7 has increased slightly over that of the previous example case and the disparity between the amplitudes of computations for different pixel pairs has grown as a result of the changed geometry.

Figure 5.7 shows that the standard slope values, although still intersecting at 0 and $\pi/2$, no longer relate to the peak and minimum values of the new slope for nearly all pixel pairs. This condition translates to Figure 5.8, where again the horizontal error to test statistic relationship is grossly overestimated by the two more conservative dual slope combinations, Max-Max Slope and Max$\{i^{th} + j^{th}\}$ Slope. The sum computations are used once again as the more conservative estimates. The singularities seen in the $Slope_{MAXMAX}$ computation in the orthogonal cases do not occur in cases $2a$ or $2b$, where all individual x and y elements making up the full set of m pixel pairs are unique. However, in case $2c$ the $Slope_{MAXMAX}$ value is once again "infinite" due to a shared x coordinate among two of the pixel pairs and a shared y coordinate among two others. This clearly demonstrates the limitation of directly applying this GPS RAIM method for multiple errors to the image-based case.

In contrast, like with the rotated case described in the previous example, the new slope formulation is tangent to the now more visible quasi-elliptical pattern, while the standard slope falls below the new slope. This again demonstrates that the new slope approach best predicts the worst-case error. Figure 5.9 provides an illustration of an increasing bias (evaluating over all angles at each bias step) showing that the new slope consistently tracks the position error based on the worst angular error condition.

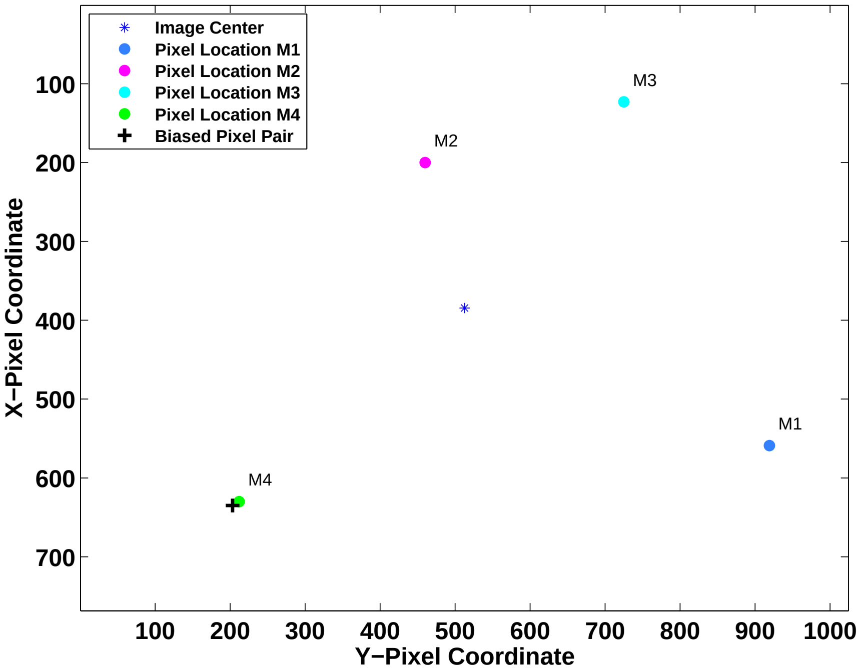

Figure 5.6: Pixel Measurement Geometry for Example 2 (Configuration $2a$).

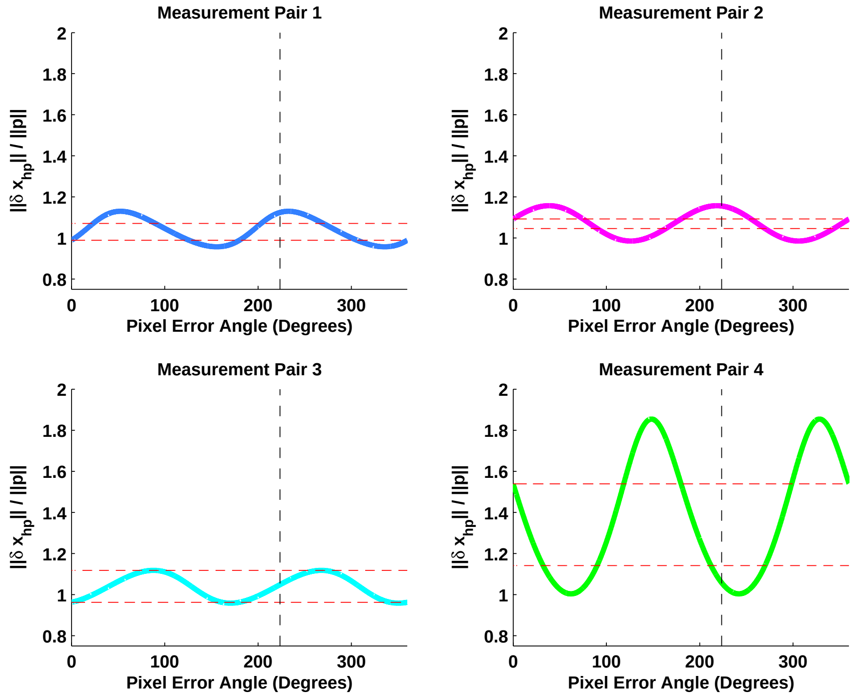

Figure 5.7: New and Standard Slope Comparison Over Varying Angles; Example 2 Asymmetric Geometry ($2a$). NOTE: The "New Slope" in each subplot is indicated by the solid line, while the "Standard Slopes," which produce the same result as the "New Slope" evaluated at both $\theta = 0$ and $\theta = \pi/2$, are indicated by the dashed lines.

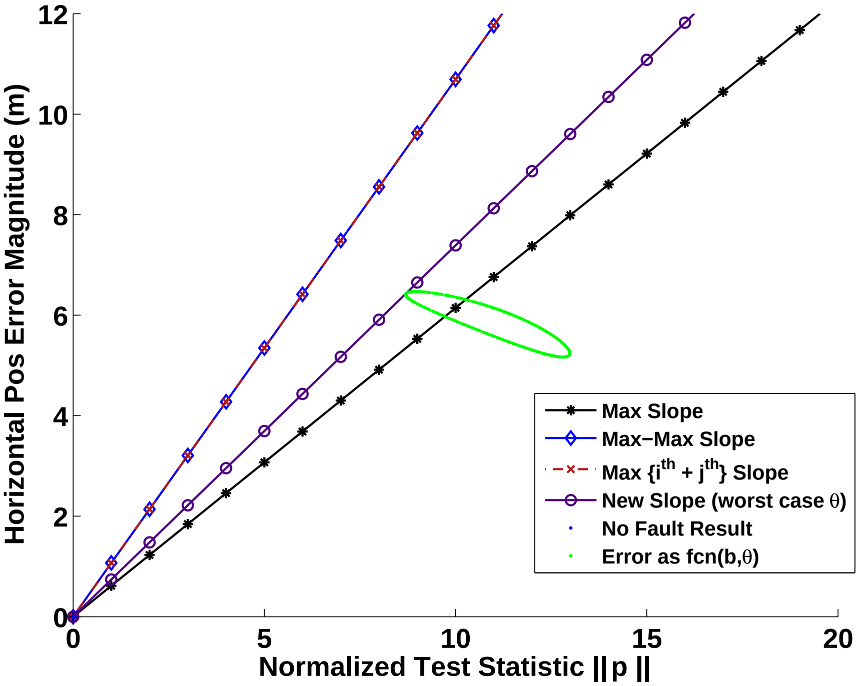

Figure 5.8: RAIM-like Slope Presentation for Comparison; Example 2 Asymmetric Geometry ($2a$).

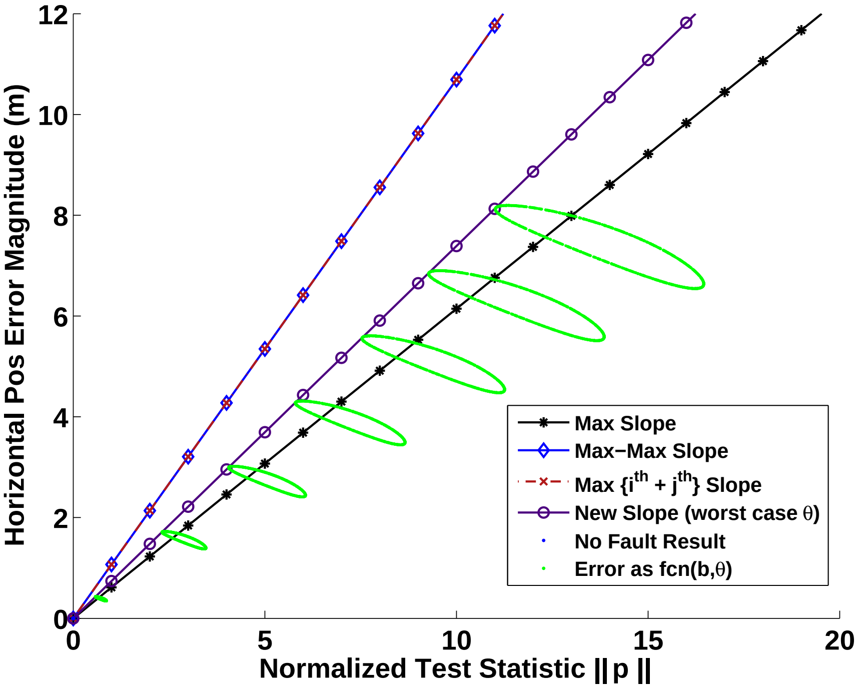

Figure 5.9: Increasing Bias Error Against New Slope; Example 2 Asymmetric Geometry ($2a$).

5.2.3 ***Extended Analysis of Position Only Case***. The influence of additive noise and subsequent adjustments to the induced bias level to satisfy probability of missed detection requirements, as well as the impact of including the attitude states, are all important considerations in the investigation of image-based integrity, and will be investigated in future work. This research focuses on the immediate concern of the behavior of the "new slope" algorithm, described in the previous sections, under changing altitude and random pixel pair locations, and the resulting impact on the horizontal position error.

Earlier analysis illustrated the deterministic horizontal position error over $[0, 2\pi)$ angular errors at a particular bias level. Remaining analysis is concentrated on the worst case angular error in conjunction with a chi-square threshold value generated using a selected probability of false alarm and the appropriate number of degrees of freedom. A deterministic horizontal position error is then computed for the point at which the respective maximum "new slope," based on the worst case error and given observation matrix, $\mathbf{H}$, intercepts a fixed bias level chosen to be equivalent to the threshold value. This definition is comparable to the Approximate Radial-Error Protected (ARP) level for GPS RAIM described in [15]. As previously mentioned, this definition is not intended to define the horizontal protection level under noisy conditions, but does provide a crisp delineation for comparing scenarios presented in this paper. Mathematically, this relationship is given by:

$$HPE_{TD} = \max_{i=1:m} [Slope_i\,] TD \tag{5.3}$$

or the horizontal position error (HPE) at threshold (TD). TD is the chi-squared threshold value determined, in this case, for the $2m - n$ degrees of freedom, with m representing the number of pixel pairs and n representing the number of states. The following evaluations, like those in the previous section, consider that only four features (four pixel pairs) are available.

Figure 5.10 shows the relationship of HPE_{TD} versus altitude, with altitude ranging from 0-3000m, for four specific geometries, each using four pixel pairs. These four baseline geometries are extensions of those presented earlier, as indicated by the numeric prefixes in the legend entries (see Table 5.1). The orthogonal configurations, given by legend entries $1a$, $1b$, and $1c$ of Figure 5.10, are also symmetric, meaning that the target pixel locations on the pixel plane are are mirrored about an imaginary line drawn through the center of the pixel plane. An additional equidistant term denotes that the relative distance, in two-dimensional magnitude, between each target's pixel location and the center of the pixel plane, is the same for all pixel pairs. The use of the rotation term further denotes that the relative distance from pixel plane center to each original pixel location is maintained while the whole configuration is rotated by an angle α about a two-dimensional coordinate axis with an origin at the center of the pixel plane. The asymmetric geometry, given by legend entry $2a$ of Figure 5.10, is exactly as shown earlier in Figure 5.6.

It is clearly seen from Figure 5.10 that the value of $HPE_{_{TD}}$ grows linearly as a function of altitude for each given geometry. Since the bias level and worst case angle values are fixed in each case, this growth trend is directly resultant from the change in each "new slope" magnitude through changes in the rows of the $\mathbf{H}$ matrix coincident with the increasing altitude. However, the differences in the rate of "new slope" growth are attributed in the differences in the baseline geometries themselves. For example, the orthogonal configurations described by $1a$ and $1b$ differ by approximately 50 pixels in distance for two out of the four pixel locations. The ratio of the slopes between these two cases is a constant 1.08 over each altitude value. When comparing the orthogonal, equidistant case, $1b$, to the asymmetric case, $2a$, this ratio grows to approximately 1.50. Finally, when comparing case $1b$ with the same configuration rotated by any angle α, as in case $1c$, the ratio is simply 1.00. This implies that an orthogonal, equidistant configuration may approach a "best-case" baseline configuration.

Having examined behavior and potential predictability under a few selected geometries, it becomes necessary to further investigate what occurs under non-predetermined geometries. Unlike in GPS scenarios, where the known satellite orbits provide a predictable geometry, the geometry of the targets in the image-based scenario cannot always be antic-

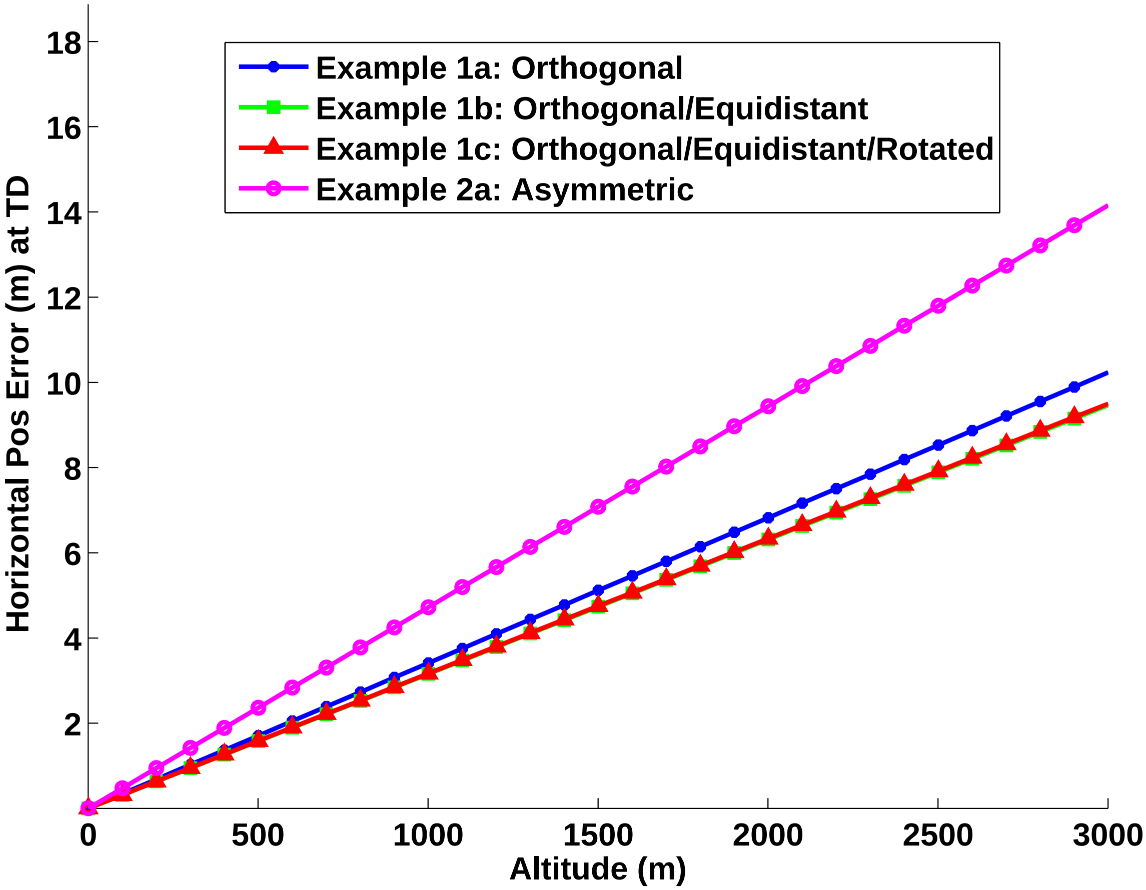

Figure 5.10: HPE_{TD} versus Altitude for Multiple Examples.

ipated, so it is helpful to statistically evaluate random pixel pair geometries. Simulations were set up using $40,000$ runs each, with each iteration using four pixel locations chosen from a random uniform distribution and the constraint that they are observable on the pixel plane and inside of the image plane border by 50 pixels. The results are compiled as a normalized histogram plotted against versus the $HPE_{_{TD}}$ term, as shown in Figure 5.11. This histogram does not immediately appear to fit a known statistical distribution. Therefore, the baseline altitude was initially set to 1000 meters to allow heuristic comparisons with the data shown in Figure 5.10.

Over the $40,000$ runs, the observed $HPE_{_{TD}}$ value ranges from a minimum of 3.31 to a maximum of 176.5 with minor variability in the minimum (± 0.3) due to bin size selection. For clarity, the x-axis in Figure 5.11 is truncated at $HPE_{_{TD}} = 60$ to better illustrate the nature of the distribution, with 0-60 chosen as the range where 99.9 percent of the samples are represented. The highest incidence of occurrence is at $HPE_{_{TD}} = 4.70$, which falls reasonably close to the 4.72 value derived from the asymmetric case, example $2a$, shown in Figure 5.10 for the $1000m$ altitude point. In fact, this same observation holds for $40,000$ runs taken at altitudes of $500m$, $1000m$, $1500m$, $2000m$, $2500m$, and $3000m$, where the largest difference in value between the histogram peak and the corresponding point taken from the data used in Figure 5.10 is 0.30, and only 0.15 on average. The minimum values also compare favorably to the baseline seen in examples $1b$ and $1c$, only slightly larger (≤ 1) in $HPE_{_{TD}}$. This would seem to imply that the orthogonal, equidistant condition is rare. Likewise, with only a small number of samples exceeding $HPE_{_{TD}} = 60$ value for $1000m$ altitude, extreme geometries are rarely encountered.

An additional analysis step was taken to further characterize the results of the histogram data used to build Figure 5.11, in order to further capture the impact of random geometries on $HPE_{_{TD}}$ determination. The data was compiled into cumulative sums versus $HPE_{_{TD}}$ to form pseudo-continuous distribution functions for altitudes ranging from $500m$ to $3000m$ in $500m$ steps, as shown in Figure 5.12. Due to the spread of the data at

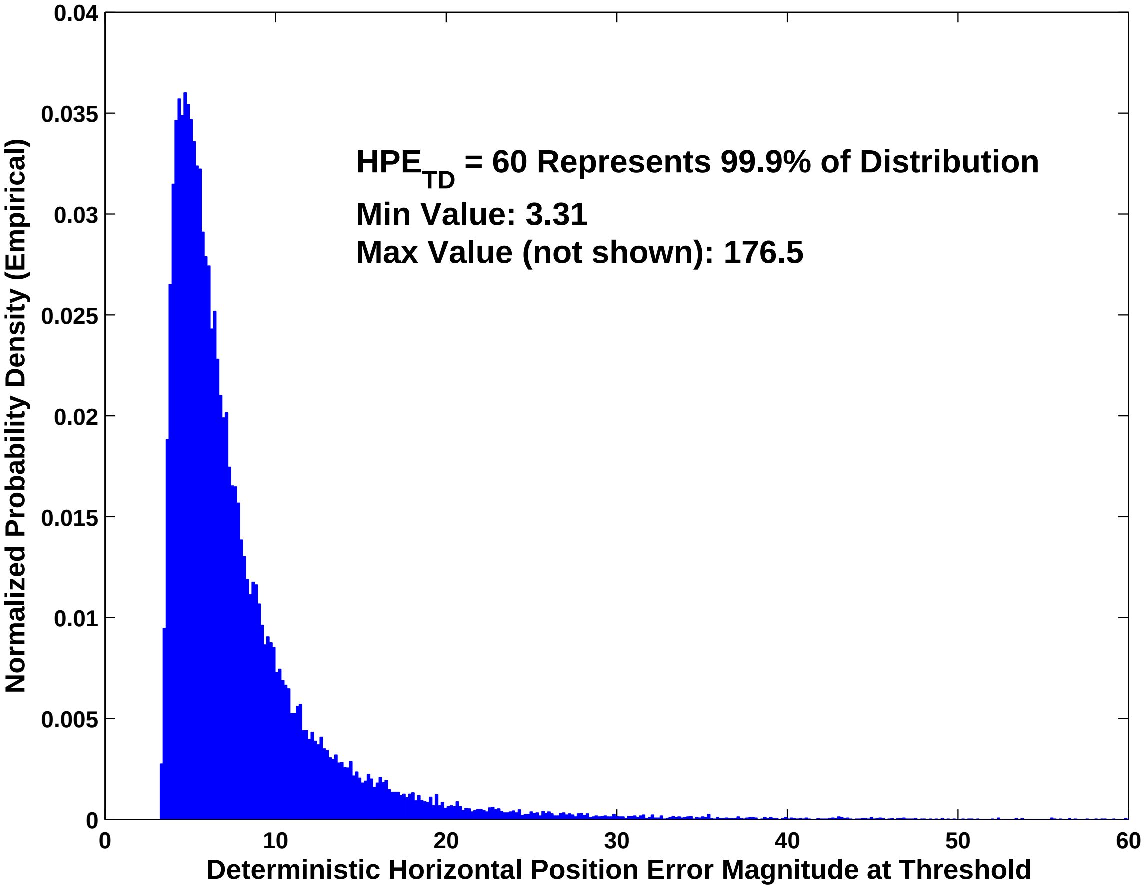

Figure 5.11: Normalized Histogram of Max New Slope x Threshold (HPE_{TD}); $40,000$ Runs with Random Geometries of Four Features.

the various altitudes, a logarithmic scale along the HPE_{TD} axis is used in order to better show the attributes of each curve. This representation clearly demonstrates the impact of altitude, since the plotted distributions shift to the right in proportion to each increase in altitude. For example, in Figure 5.12, the 99.9 percent point described earlier for $1000m$ altitude actually occurs at $HPE_{TD} = 58.53$. In comparison, increasing the altitude to $2000m$ then $3000m$ results in 99.9 percent HPE_{TD} values of 115.2 and 172.8, respectively. This plot consequently reinforces the importance of altitude in the determination of a horizontal protection level for image-based navigation.

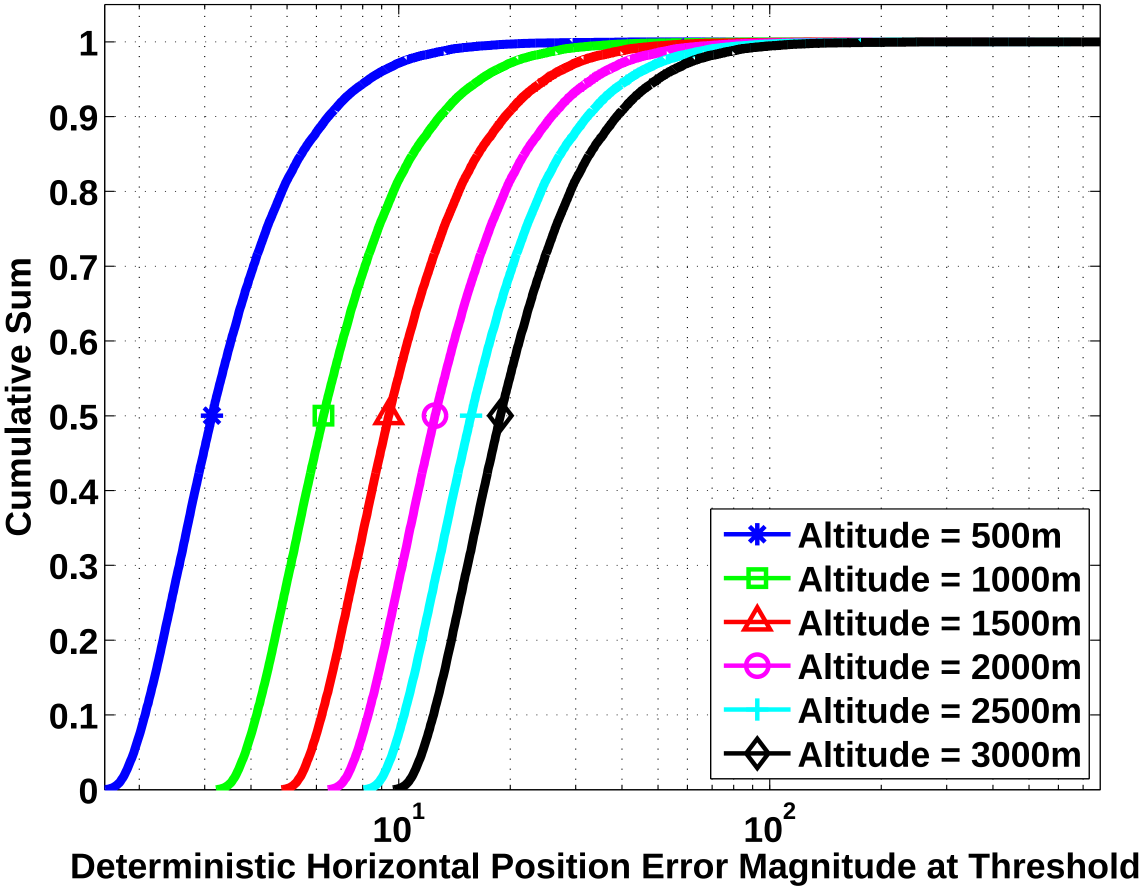

Figure 5.12: Cumulative Sum of Histogram Values, Random Geometries.

One final observation in Figure 5.12 is that these cumulative sums based on the empirical data indicate a "zero probability" below the minimum value observed in the histogram. This effect is the result of the fixed threshold value used in computing $HPE_{_{TD}}$, as seen from Equation (5.3). Thus, although the generation of the random geometries may produce a near optimal configuration, or "best case" geometry, under this condition the realization of the minimum value will not be reach zero for a non-zero threshold.

5.3 Estimating both Position and Attitude

In this section, the scenario of estimating position only is extended to investigate the impact of adding attitude states to the position error vector, $\delta\mathbf{x}$, and properly accounting for the new states in the observation matrix, $\mathbf{H}$. In this case, $\delta\mathbf{x}$ and $\mathbf{H}$ are exactly as described in Equations (4.3) and (4.5), respectively. Incorporation of the new elements in both the error state vector and observation matrix do not alter the slope relationship previously derived in Equation (4.17), but the inclusion of the attitude states has a significant impact on slope results, as will be shown in the following analysis.

This investigation is done under noise-free conditions, so the position error and test statistics generated in simulation are deterministic. This type of evaluation allows a clearer demonstration of the relationship between the horizontal position error magnitude and the potential observable. The $HPE_{_{TD}}$ metric given in Section 5.2 is again used for performance comparisons, with TD as the chi-squared threshold value established as the bias point. Comparisons in this case are taken between the estimating position only and estimating position plus attitude scenarios, since GPS RAIM does not commonly take into account the attitude states and a detailed comparison between GPS RAIM and the case where only position is estimated was accomplished in the last section.

As described in Section 2.13.1, the threshold value is a function of both the number of degrees of freedom and the probability of false alarm, p_{fa}. Changing the specification on this probability changes the corresponding threshold value, with a reduction in p_{fa} increasing TD and an increase in p_{fa} reducing the value of TD. The following evaluations initially consider that only four pixel pairs are available at a fixed probability of false alarm. Therefore, only the number of degrees of freedom differ between the two estimation cases, and drive the different threshold values. Figure 5.13 illustrates this condition.

The fact that two rows of the $\mathbf{H}$ matrix are required for each pixel pair makes direct comparisons based on the number of degrees of freedom impossible, since the number of degrees of freedom will always be an odd number for the position only case and an even number in the combined position plus attitude case. In practice, comparison based on this metric would be impractical, even if possible, since the solutions are driven by the number of available pixel pairs. Figure 5.13 clearly shows a difference between the slope based on estimating position only and that based on estimating both position and attitude, and this difference is the main focus of the following discussions.

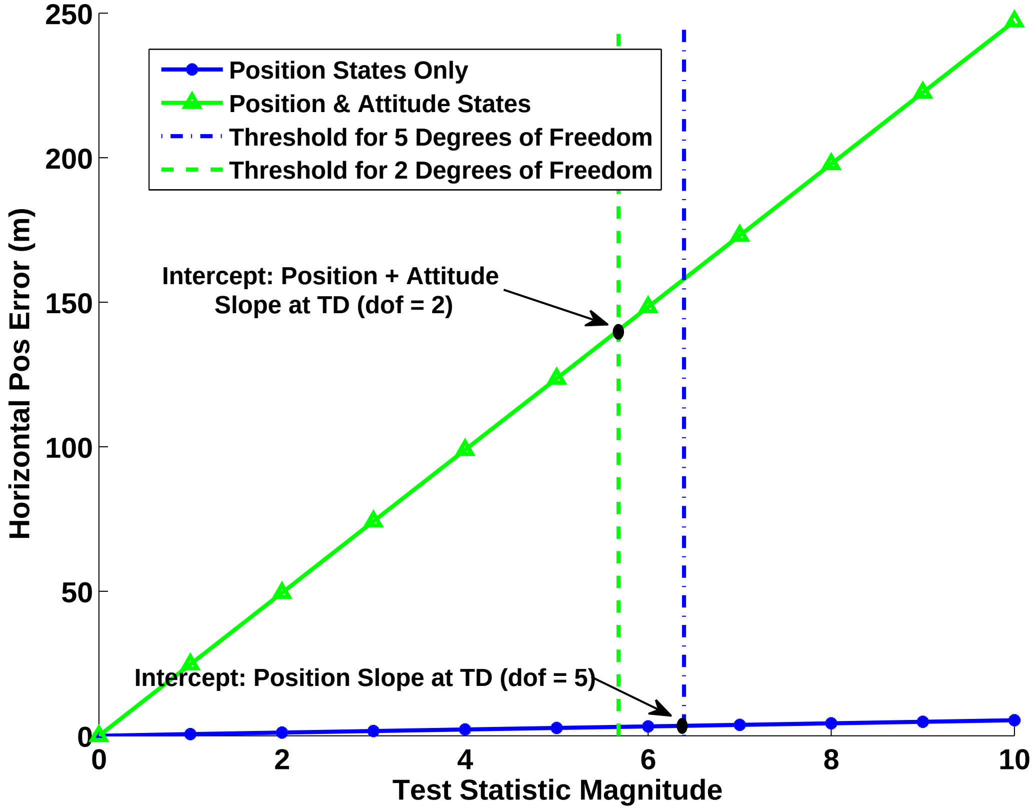

Figure 5.13: Illustration of Slope Intercept at Threshold (TD).

5.3.1 ***Setting Position Only Baseline***. For the analysis that follows, a baseline set of four pixel location geometries, each using four pixel pairs, is used. These four baseline geometries are extensions of those presented in detail in Section 5.2 and are summarized here.

There are three base orthogonal configurations. These configurations are symmetric, meaning that the target pixel locations on the pixel plane are mirrored about an imaginary line drawn through the center of the pixel plane. An additional equidistant term denotes that the relative distance, in two-dimensional magnitude, between each target's pixel location and the center of the pixel plane, is the same for all pixel pairs. The use of the rotation term further denotes that the relative distance from pixel plane center to each original pixel location is maintained while the whole configuration is rotated by an angle α about a two-dimensional coordinate axis with an origin at the center of the pixel plane. The Figure 5.13 example is based on a specific four pixel pair orthogonal, equidistant geometry like that shown in Figure 5.14.

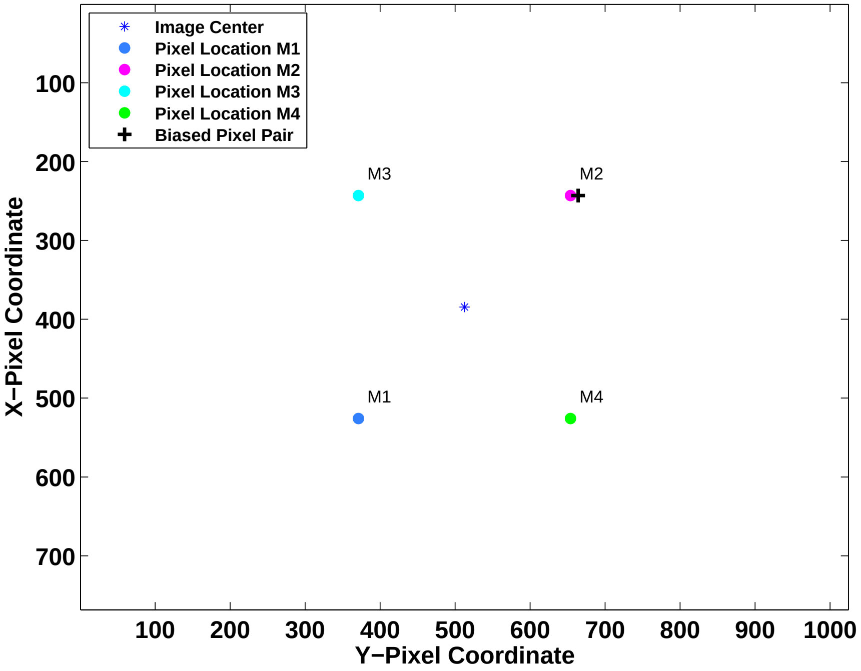

Figure 5.14: Pixel Measurement Geometry; Symmetric (Configuration $1c$).

The fourth base configuration is asymmetric, as previously shown in Figure 5.6. Though limitless possible target geometries exist, this asymmetric example is chosen here to represent a reasonable condition where the four pixel pairs are located roughly in four quadrants of the image plane. Although no longer orthogonal or symmetric in distribution, the configuration would seem reasonable for admission into a potential position solution based on geometric criteria, and thus provides an example for comparison.

The magnitude of the "new slope" under the worst angular error condition is strongly influenced by altitude through the formulation of $\mathbf{H}$ and the subsequent pseudoinverse operations. As the distance between the target and the camera platform grows, so too does the peak value of the "new slope." A comparison is shown through Figure 5.15 and Figure 5.16, which are evaluations of the slope algorithm in the asymmetric geometry for the position only case. These evaluations are taken over all values of possible angular errors, at altitudes of 1000m and 2500m, respectively, with the latter replicating Figure 5.7.

Noting the difference in scale, the ratios of the respective slopes for each pixel pair at the two different altitudes is found to be equivalent to the ratios of the altitudes themselves, indicating a linear relationship. This condition is even more evident in the symmetric geometries. It should be noted that the straight lines shown across each of the subplots reflects the conventional RAIM slope formulation for comparison, and to observe that the "new slope" is sensitive to changes in the angular error, while the basic RAIM model is not.

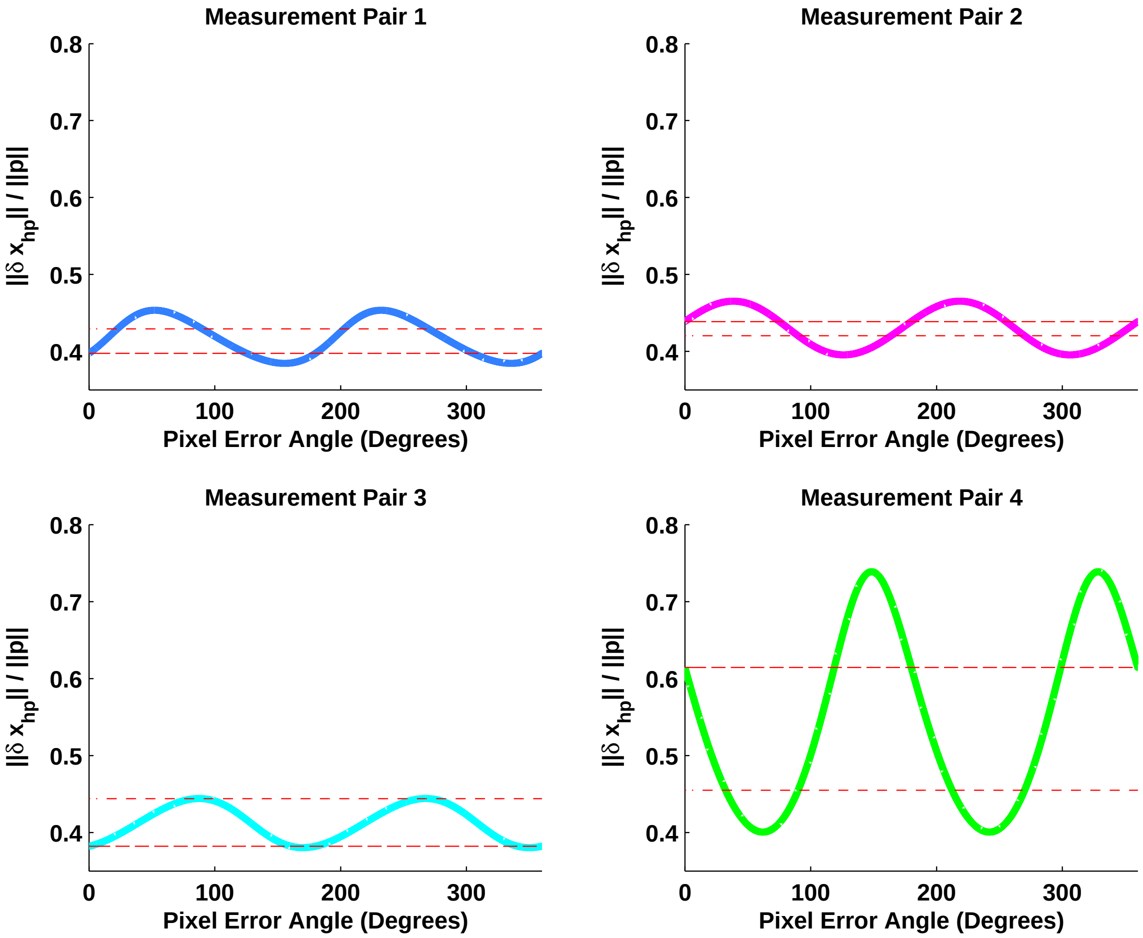

Figure 5.15: Slope Comparison Over Varying Angles in Estimating Position Only; Asymmetric Geometry $(2a)$ at 1000m Altitude. NOTE: The “New Slope” in each subplot is indicated by the solid line, while the “Standard Slopes,” which produce the same result as the “New Slope” evaluated at both $\theta = 0$ and $\theta = \pi/2$, are indicated by the dashed lines.

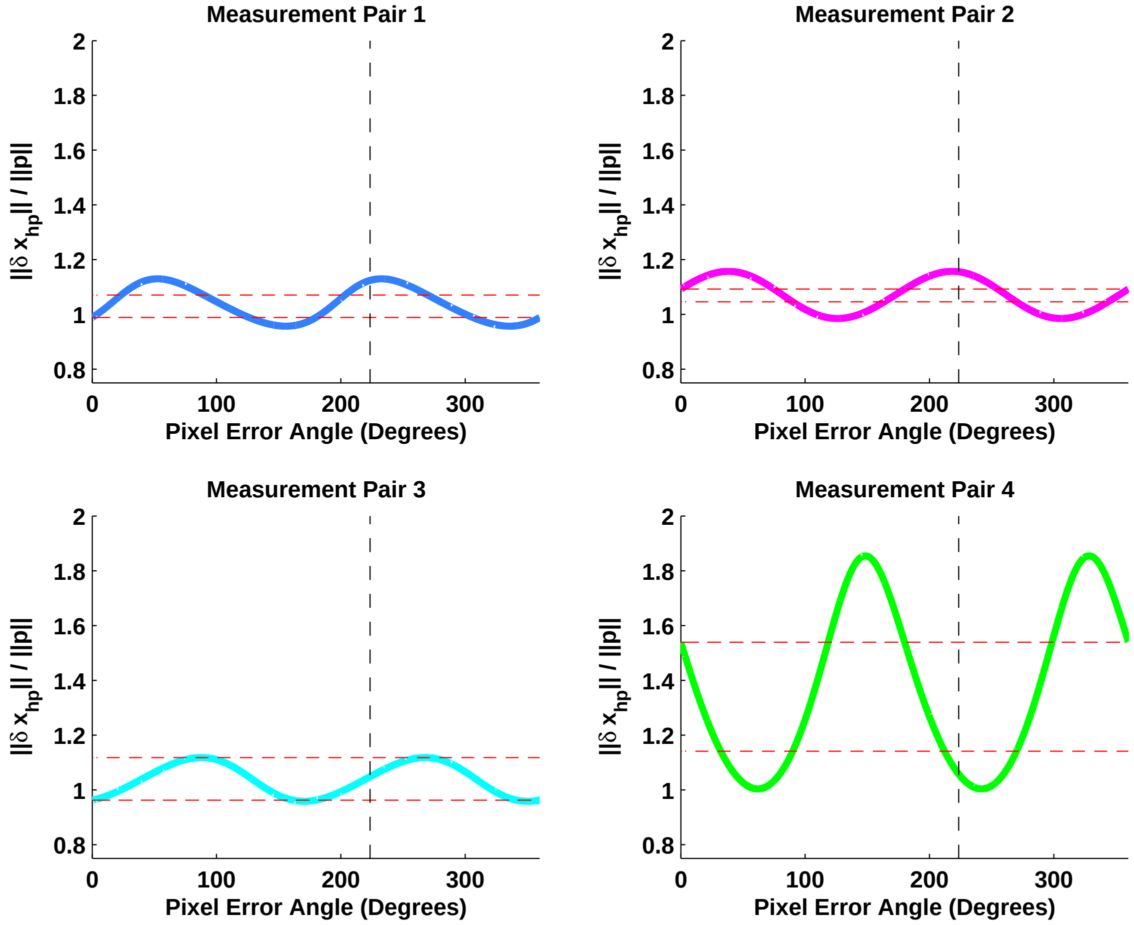

Figure 5.16: Slope Comparison Over Varying Angles in Estimating Position Only; Asymmetric Geometry $(2a)$ at 2500m Altitude. NOTE: The "New Slope" in each subplot is indicated by the solid line, while the "Standard Slopes," which produce the same result as the "New Slope" evaluated at both $\theta = 0$ and $\theta = \pi/2$, are indicated by the dashed lines.

5.3.2 Estimating Position and Position Plus Attitude.

These observation led to the comparisons between the slope based on position only and that based on both the position and attitude states being taken over an altitude range of 0m to 3000m. Figure 5.17 shows the relationship of $HPE_{_{TD}}$ versus altitude for the position only case in the four specific geometries, each using four pixel pairs. The three orthogonal configurations are given by legend entries $1a$, $1b$, and $1c$, and are further identified as equidistant and/or rotated. The asymmetric geometry, given by legend entry $2a$ of Figure 5.17, is exactly as shown earlier in Figure 5.6 (for pixel coordinate association with legend entry see Table 5.1). Figure 5.18 provides an equivalent representation, where now the slopes are based on the both the position and attitude states incorporated into the slope algorithm.

Figure 5.17 reprises the estimating position only versus altitude plot shown earlier in Section 5.2.3. Recall from that section that the ratio of the trajectories between the orthogonal configurations described by lines $1a$ and $1b$ was a constant 1.08 over each altitude value in the position only case. The ratio when comparing the orthogonal, equidistant geometry, $1b$, to the asymmetric geometry, $2a$, rose to approximately 1.50. Finally, the comparison between the $1b$ line with the same configuration rotated by any angle α, as in line $1c$, yielded a ratio of 1.00.

Similar relationships also can also be seen in the position plus attitude case using the same geometries, as shown in Figure 5.18. In this illustration, the relative comparisons between the different geometries are on the same order as those seen in the position only case. Specifically, the ratio between the geometries represented by lines $1a$ and $1b$ show a ratio of 1.27, while the ratio between the asymmetric and perfectly symmetric geometries, lines $2a$ and $1b$, respectively, is 1.98.

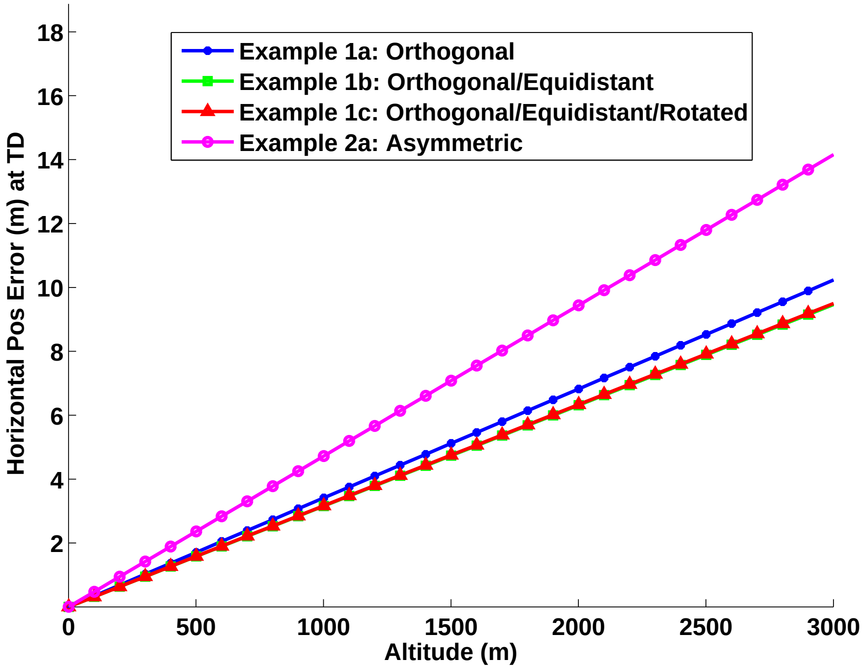

Figure 5.17: Comparison of Slopes, When Estimating Position Only.

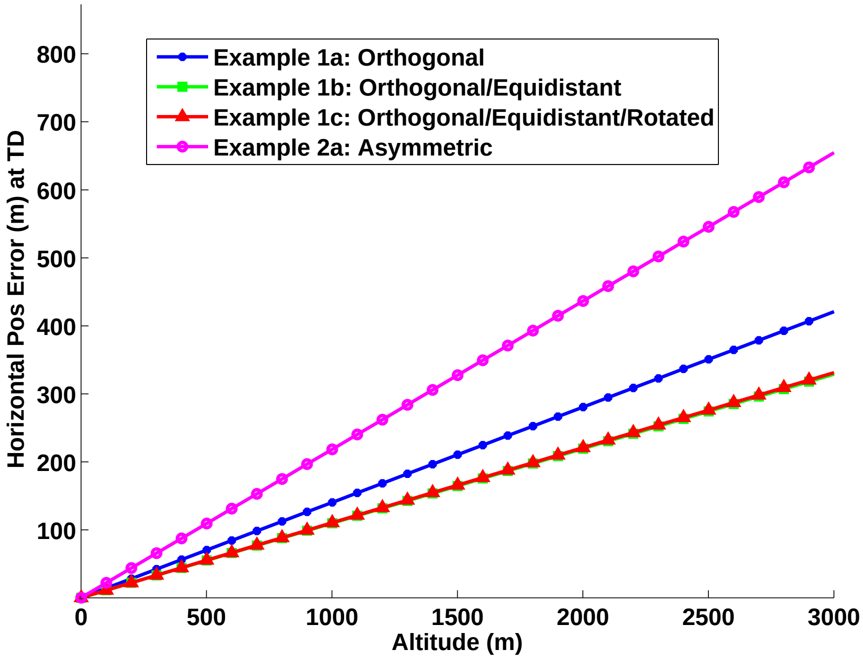

Figure 5.18: Comparison of Slopes, When Estimating Position plus Attitude (Note Increase in Scale Over Figure 5.17).

A significant result is seen when comparing the two cases, estimating position only and estimating position plus attitude, across common geometries. When comparing the asymmetric geometry in the two cases, the position plus attitude result is a constant 46.24 times larger than the position only result. Although the two case use a different TD value to formulate the horizontal position error at threshold, the change in TD, due to a differing number of degrees of freedom, only accounts for about 1 percent of the difference and, if it were the only factor, would actually reduce the difference between the two cases. Instead, the key contributor to the 46.24 factor difference is the change in the new slope computation when estimating both position and attitude. Figure 5.19 shows the new slope evaluations over all angles for each pixel pair in this asymmetric geometry, now taking into account the addition of the attitude states.

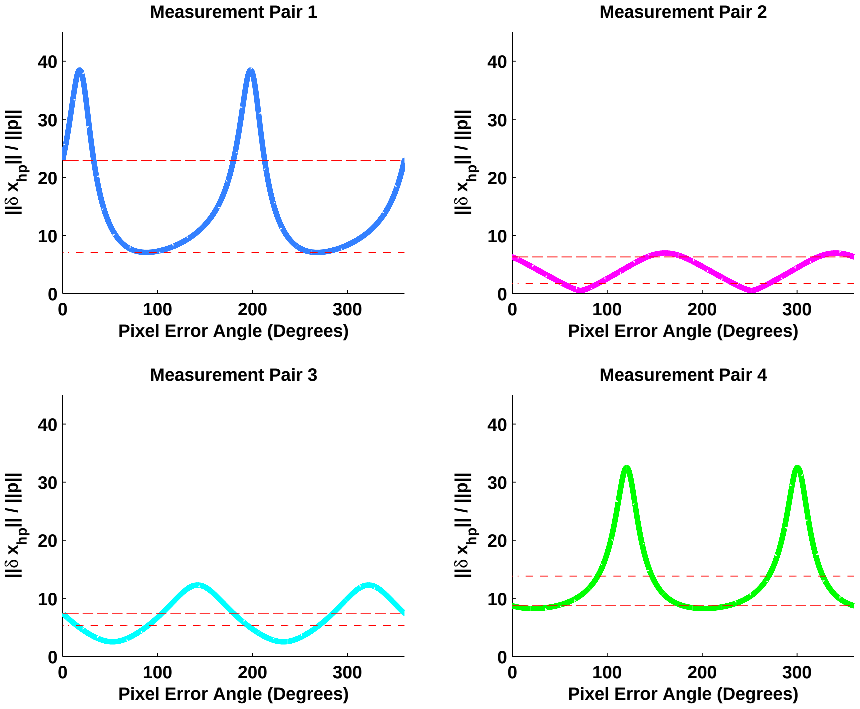

Figure 5.19: Slope Comparison Over Varying Angles in Estimating Position and Attitude; Asymmetric Geometry ($2a$) at 1000m Altitude. NOTE: The “New Slope” in each subplot is indicated by the solid line, while the “Standard Slopes,” which produce the same result as the “New Slope” evaluated at both $\theta = 0$ and $\theta = \pi/2$, are indicated by the dashed lines.

Examination of Figure 5.19 reveals that there is now a pronounced peaking that is occurring in what were formerly more sinusoidal new slope evaluations shown earlier in Figure 5.15 for the estimating position only case. Figure 5.19 also shows that the addition of the attitude states can change the determination of which pixel pair has the greatest influence on the integrity calculation, where in this case the maximum new slope value is found in the evaluation of pixel pair 1, while in the position only case the maximum value is found in the evaluation of pixel pair 4. Since the new slope approach is extracting the largest value of any pixel pair corresponding to the worst case error in terms of an angular anomaly, the differences in the trajectories observed between Figure 5.17 and Figure 5.18 should be expected based on this observation. Similar results occur when comparing the two estimation case in orthogonal geometries. In the symmetric, equidistant geometries, the position plus attitude amplification is 34.87 over the position only case, while the simple orthogonal geometry result is 41.14 times greater in the estimating position plus attitude case than in the estimating position only case.

The underlying cause for the dramatic change between the cases is attributed to the addition of the columns associated with the attitude states being added to the $\mathbf{H}$. These changes in $\mathbf{H}$ amplify the magnitude of the $\mathbf{H}$ matrix in the numerator of Equation (4.18), since the construct contains new columns of terms greater than 1. At the same time, the magnitude of the denominator, a function of the Parity matrix, $\mathbf{P}$, decreases slightly. Geometry plays a key role in the final computation due to what can be called a phasing scenario. In orthogonal geometry, the denominator remains relatively constant over all angles in the new slope, but under asymmetric geometry, the denominator exhibits a more pronounced sinusoidal behavior, in some cases directly out of phase with the numerator, thus amplifying the overall ratio and creating the peaking effect. This clearly demonstrates the impact of estimating attitude on any evaluation of a horizontal protection level.

Evaluations comparing the new slope algorithm developed in Chapter IV with the basic RAIM Slope, Max-Max Slope and Max$\{i^{th} + j^{th}\}$ Slope are also done for the new estimating both position and attitude scenario. Aside from the increase in slope magnitude across the board, the results are consistent with those described in Section 5.2, with one notable exception. It was found that when adding the attitude states, the Max$\{i^{th}$ + $j^{th}\}$ Slope derived from [17] yields the same value as the proposed new slope formulation. However, this observation only appears to be true in the four pixel pair scenario, where there are $2m-n=2$ degrees of freedom. When more pixel pairs are added, the Max$\{i^{th}$ + $j^{th}\}$ Slope once again becomes very conservative while the new slope algorithm remains tangent to the position error to test statistic ratio at the worst-case angular value.

Having examined behavior and potential predictability under a few selected geometries, it becomes necessary to further investigate what occurs under non-predetermined geometries. Simulations were set up using $40,000$ runs each, with each iteration using pixel locations chosen from a random uniform distribution and the constraint that they are observable on the pixel plane and inside of the image plane border by 50 pixels. Simulated data was taken for four different redundancy conditions, using 4, 6, 8 and 10 pixel pairs.

The results are compiled as a normalized density plotted against the $HPE_{_{TD}}$ term, as shown in Figure 5.20 and Figure 5.21 for the position only and position plus attitude cases, respectively. These histograms do not immediately appear to fit a known statistical distribution. Therefore, the baseline altitude was initially set to 1000 meters to allow heuristic comparisons with the data shown in Figure 5.17 and Figure 5.18. The direct comparisons are made only to the first subplot, which in each of the two error-state cases is the representation of the four pixel pair condition and the condition upon which the earlier geometry comparisons were drawn.

In the position only case, a new set of data is taken based on the same parameters as in Section 5.2.3. Over the $40,000$ runs, the observed $HPE_{_{TD}}$ value ranges from a minimum of 3.24 to a maximum of 259.36, a maximum much larger than the 176.5 value observed in the previous test. In order to clearly show the distribution in this first subplot in Figure 5.20, the distribution plot is again truncated at the value corresponding to the 99.9 percent of the distribution, which falls at a value of $HPE_{_{TD}} = 57$, compared to $HPE_{_{TD}} = 60$ earlier. So, although the value of the maximum outlier jumped by a signif-

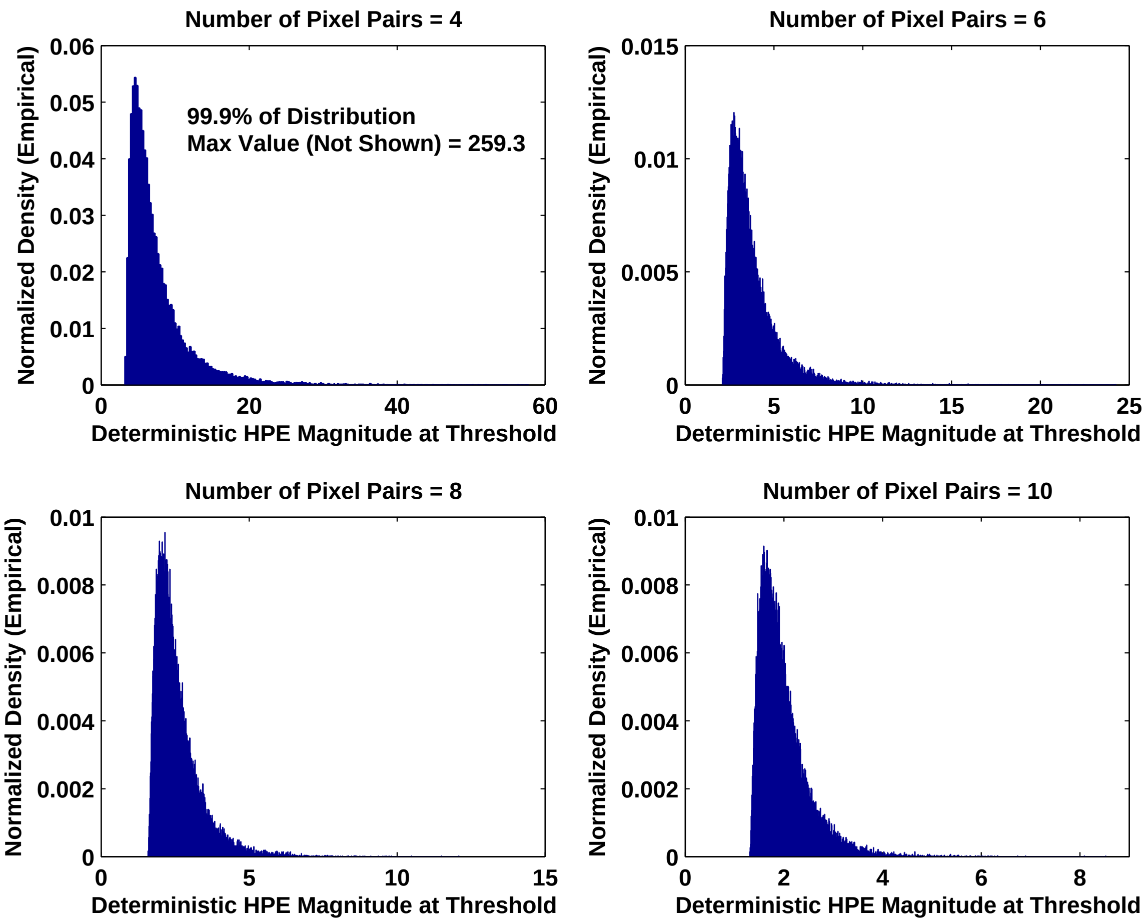

Figure 5.20: Normalized Densities vs. HPE at Threshold; When Estimating Position Only.

icant margin, the cut-off limit containing 99.9 percent of the samples has stayed relatively consistent. The highest incidence of occurrence is at $HPE_{_{TD}} = 4.62$, which falls reasonably close to the 4.72 value derived from the asymmetric case, example $2a$, shown in Figure 5.17 for the $1000m$ altitude point. In fact, this same observation holds for $40,000$ runs taken at altitudes of $500m$, $2000m$, and $2500m$, where the largest difference in value between the histogram peak and the corresponding point taken from the data used in Figure 5.17 is 0.22, or 0.16 on average. Again, the minimum values also compare favorably to the baseline seen in examples $1b$ and $1c$, only slightly larger (≤ 1) in $HPE_{_{TD}}$. This would seem to imply that the orthogonal, equidistant condition is rare. Similarly, extreme geometries also appear rare from this data, with only small number of samples exceeding the new $HPE_{_{TD}}$ value.

The additional subplots clearly demonstrate the impact of adding additional pixel pairs to the computations. As the redundancy increases, the distribution tightens with fewer and fewer large outliers persisting. In addition, although the geometries are random, the addition of more and more pixel locations increase the likelihood of better geometries. As a result, even with a possible increase in the threshold value due to changes in the number of degrees of freedom, the magnitude of the worst case slope can decrease resulting in a net reduction in $HPE_{_{TD}}$ values overall.

The distributions for the position plus attitude case, shown in Figure 5.21 take on a similar appearance to those of the position only case. However, the addition of the attitude states, which have been shown to dramatically increase the "new slope" value, and the large sample population create an incredibly wide dispersion of $HPE_{_{TD}}$ possible values. For example, although the distribution is shaped like that in the position only estimation case, the range of values in this four pixel pair scenario ranges on the orders of 10^1 to 10^7. In order to clearly present the data, the x-axis for the subplots in Figure 5.21 have been plotted in log scale. Again, the comparison with Figure 5.18 applies only to the first plot.

The most frequently occurring value in the four pixel pair subplot of Figure 5.21 is estimated at 194, which compares reasonably well to the 218 value taken at 1000m altitude in the group plot versus all altitudes of Figure 5.18. The minimum value is found to be close to 33.84 which is significantly lower than that estimated for the symmetric, equidistant line given in Figure 5.18 for 1000. Some degree of error is attributed to both the requirement for coarser discretization of the histogram to cover such a broad range of values and the higher degree of variability the added attitude states bring to the computation, as previously discussed.

It also clear that a large number of random geometries create extremely large values in $HPE_{_{TD}}$, so large in fact that a horizontal protection level defined in these extremes would not likely serve meaningful applications. The subsequent subplots in Figure 5.21,

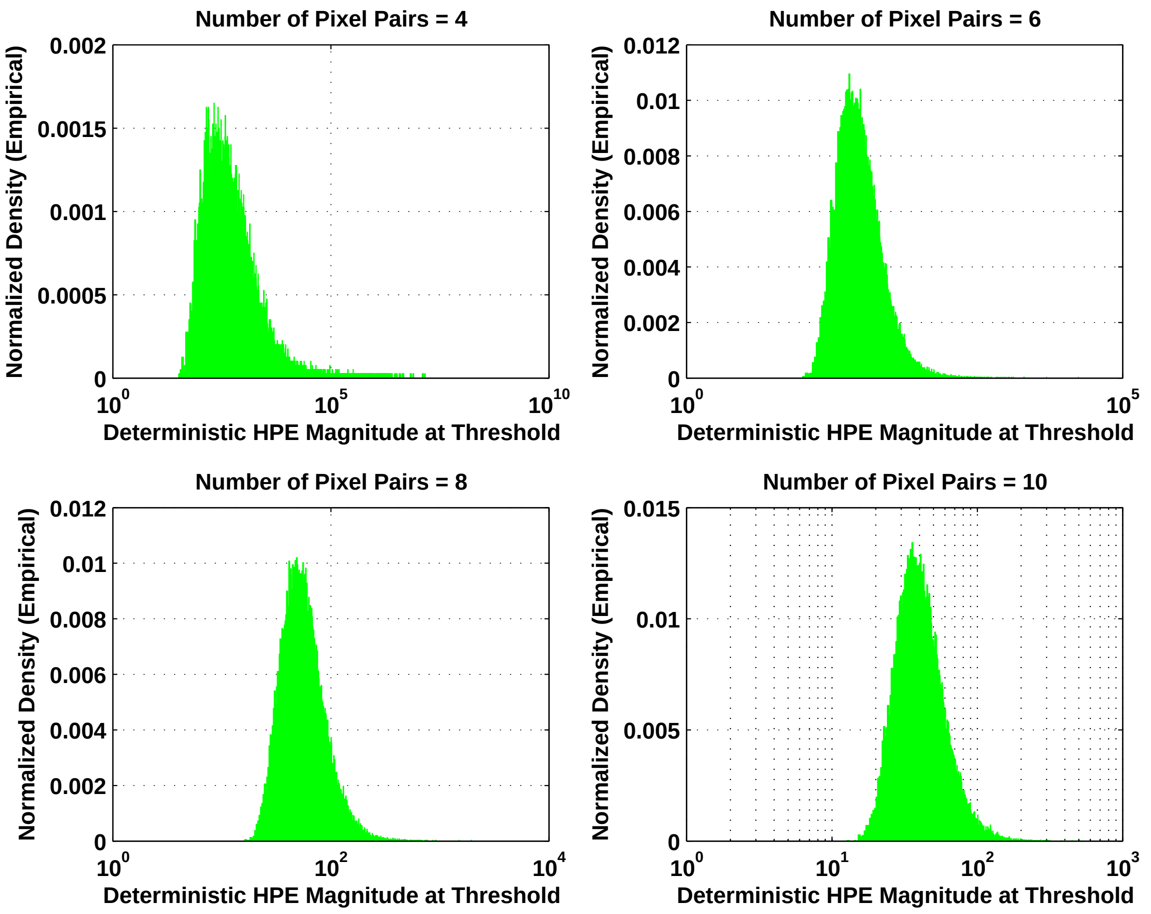

Figure 5.21: Normalized Densities vs. HPE at Threshold; When Estimating Position and Attitude (Note Log Scale).

each considering additional pixel pairs, show once again that increasing the number of pixel pairs drastically reduces the magnitude of outliers, in this case bringing the order of magnitude of the extreme values down by approximately four orders of magnitude and into the hundred meter range. Even in this case, there is a very significant benefit to knowing attitude to minimize the need for error estimation.

An additional analysis step was taken to further characterize the results of the histogram data used to build normalized density plots, in order to further capture the impact of random geometries on HPE_{TD} determination. The data was compiled into cumulative sums versus HPE_{TD} to form sets of pseudo-continuous distributions functions, as shown in Figure 5.22 through Figure 5.25 for the 500m, 1000m, 1500m, 2000m, 2500m and 3000m altitude results.

Both error-state configurations are represented in the series of plots, with examples shown for four and eight pixel pair assumptions. Plotted in log scale along the x-axis, these cumulative sums based on the empirical data indicate a “zero probability” below the minimum value observed in the histogram. This is consistent with the idea that there is no perfect geometry, only a minimum baseline. These plots once again reinforce the impact of altitude on the determination of a horizontal protection level for image-based navigation.

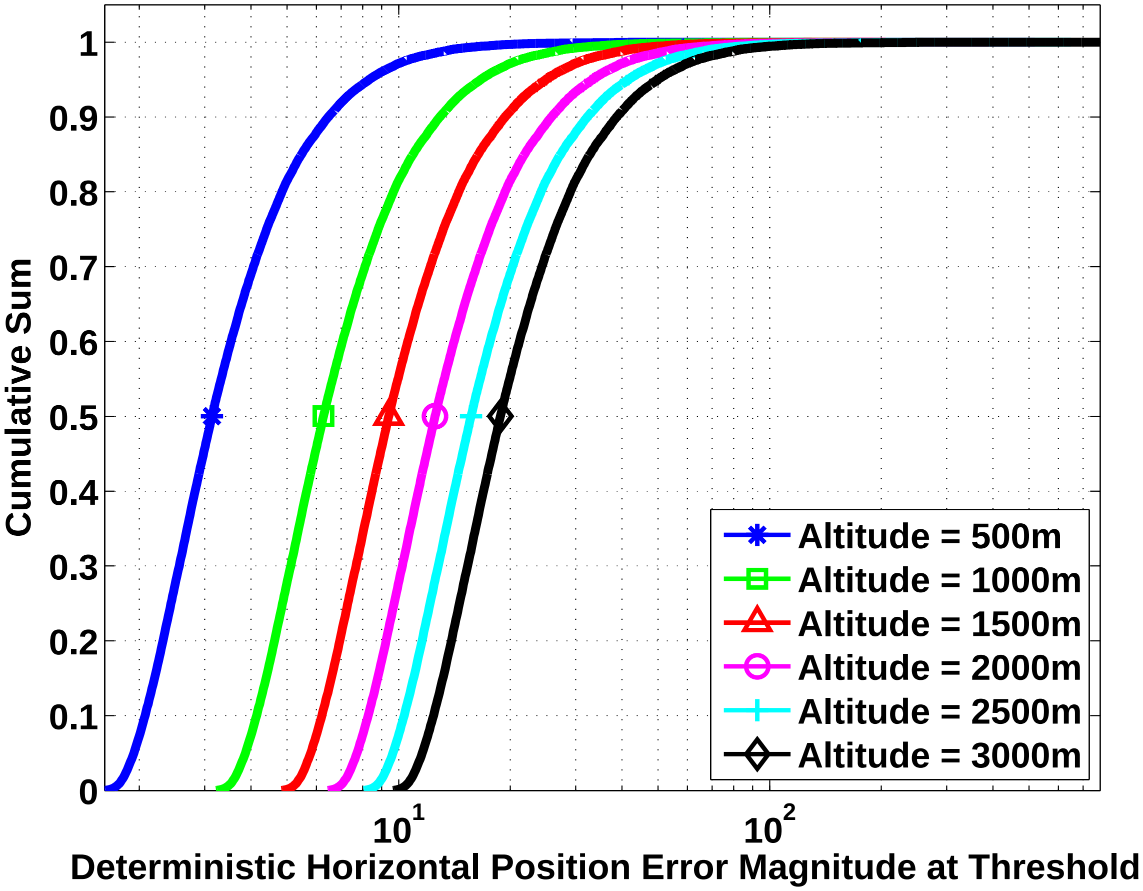

Figure 5.22: Cumulative Densities Using 4 Measurements; When Estimating Position Only.

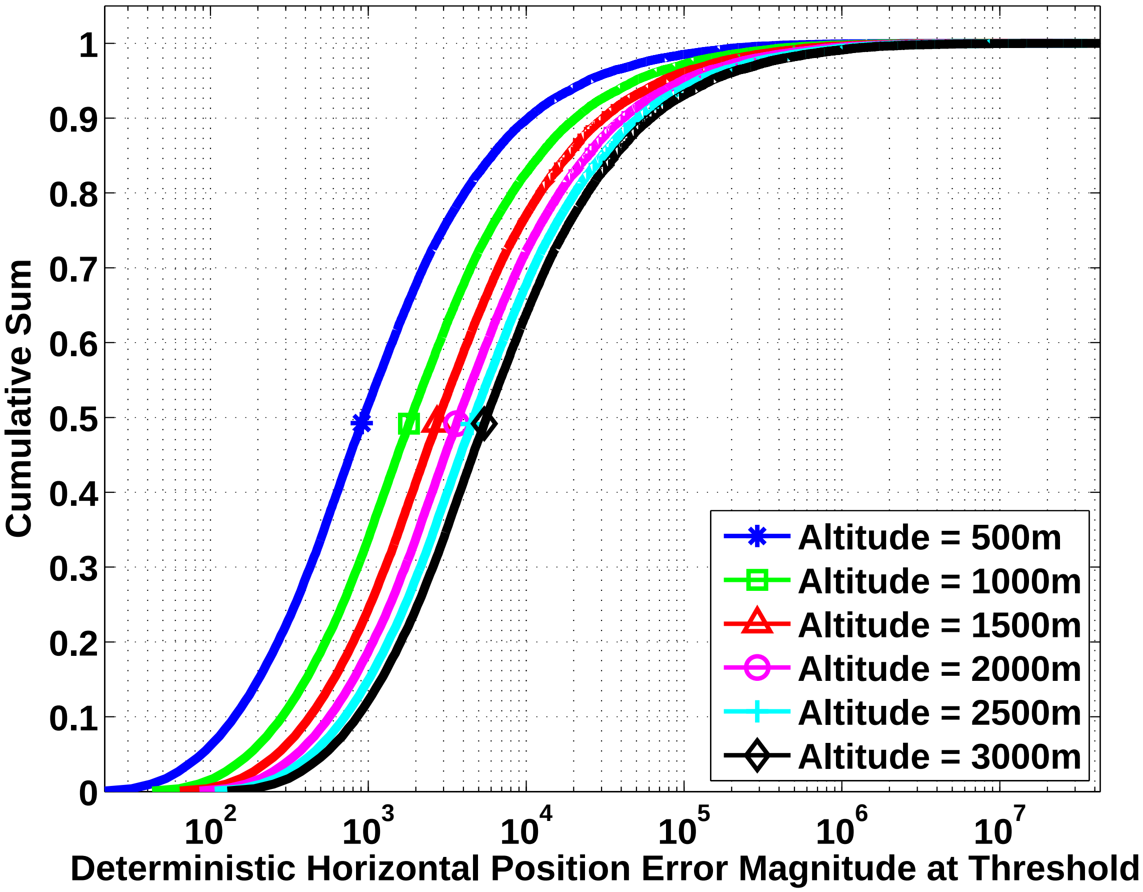

Figure 5.23: Cumulative Densities Using 4 Measurements; When Estimating Position and Attitude States.

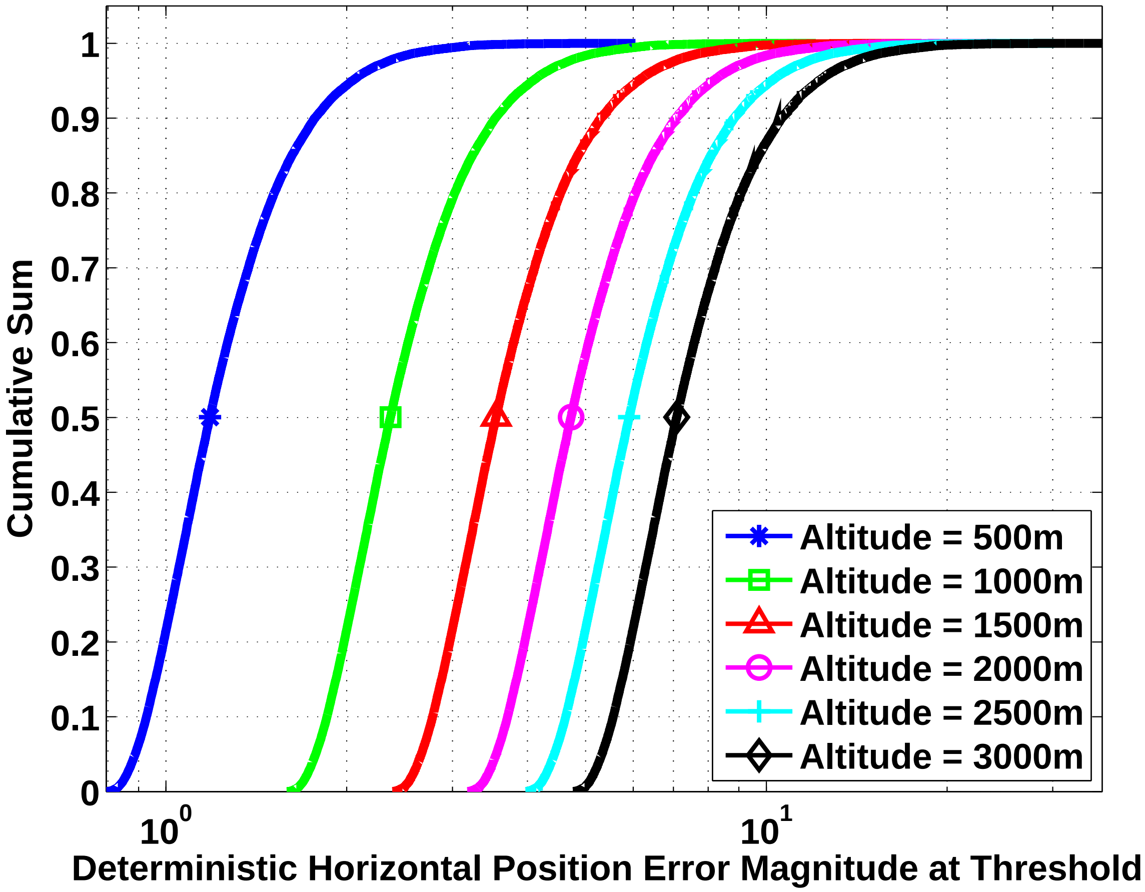

Figure 5.24: Cumulative Densities Using 8 Measurements; When Estimating Position Only.

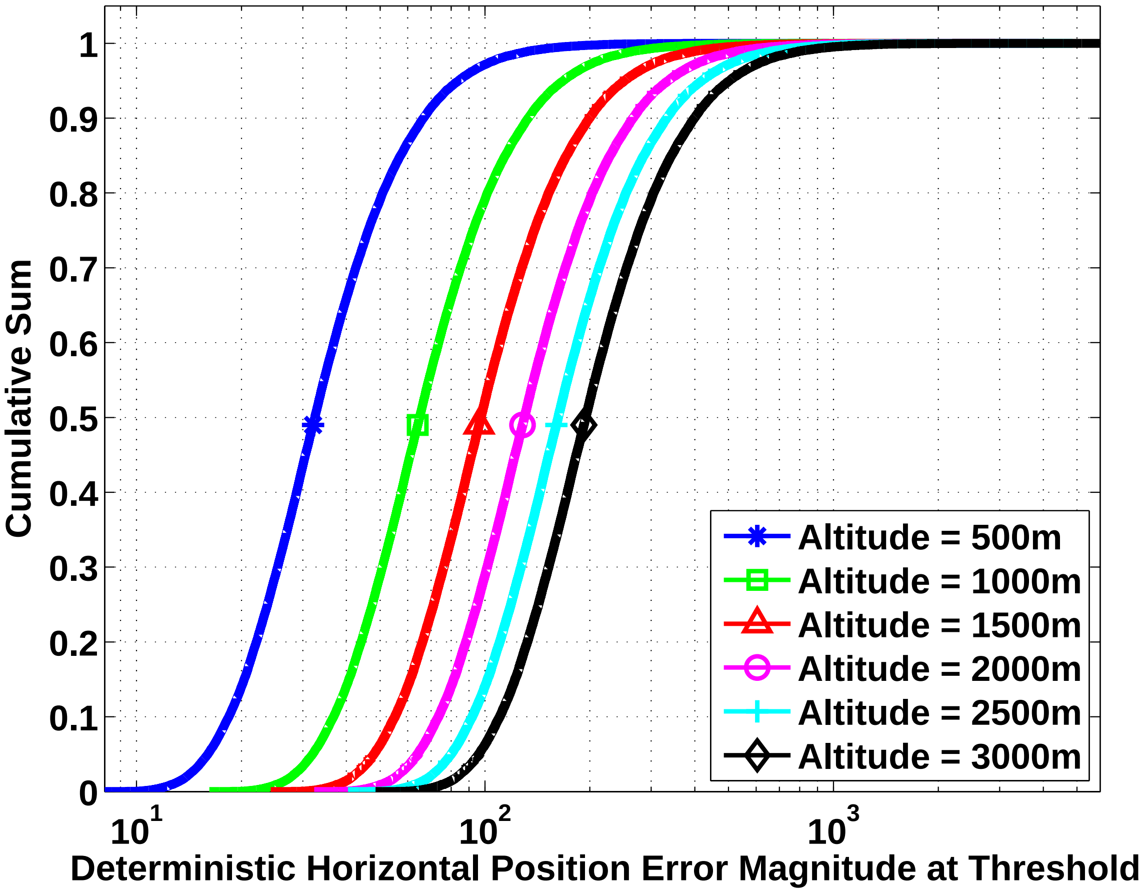

Figure 5.25: Cumulative Densities Using 8 Measurements; When Estimating Position and Attitude States.

5.4 Influence of Geometry on Image-Based Integrity

The results presented in the last section demonstrate a consistency in data structure that emerges from histograms compiled for HPE_{TD} values evaluated over large sample sets of possible pixel locations. These results also clearly show the benefit of increasing redundancy in reducing the spread of the potential outcomes and illustrate the potential for improvement in image-based integrity by limiting the maximum HPE_{TD} value that may be observed. These observations motivate the additional exploration into the potential to improve image-based integrity through a focused approach to increasing redundancy described in this section.

In the following analysis, specific four pixel pair cases taken from Table 5.1 are used to form baseline geometries, and the worst-case slope, denoted $\max_{i=1:4}[Slope_i]$, is calculated. Simulations are then run adding a fifth target at pixel location (x_j, y_k) for $j = 1 : M, k = 1 : N$ in p pixel steps, where M is the maximum number of pixels in the x-pixel plane direction and N is the maximum number of pixels in the y-pixel plane direction. A new worst-case slope for the five pixel pair case, $\max_{i=1:5}[Slope_i]$, is computed at each step. This evaluation is motivated by an interest in finding ways to improve image-based integrity not just by adding an additional target to increase redundancy, but also determining specific added target locations that would provide the greatest benefit. This knowledge could aid image-based algorithms in selecting groups of targets based on their integrity potential.

Composite results for estimating position only, followed by the same geometry evaluated for estimating both position and attitude, for two orthogonal cases and all three asymmetric cases are shown in Figures 5.26 through 5.33. As described in the previous paragraph, each subplot in these figures is a function of the location of the added, fifth, pixel pair with an evaluation done at each new location and the results plotted over the full image-plane dimensions. In each of the figures, the top left subplot is the $HPE_{_{TD}}$ metric evaluated for the five pixel pair case. For comparison, the top right subplot is the $HPE_{_{TD}}$ metric evaluated for the five pixel pair case normalized by the four pixel pair worst-case slope value, $\max_{i=1:4}[Slope_i]$ times the five pixel pair case threshold value. The intent of the normalization is to illuminate the amount of improvement resulting from the added pixel pair. Using the same TD value, based on a fixed p_{fa} and the number of degrees of freedom in the five pixel pair case, ultimately reduces the comparison to a ratio of the worst-case slopes for the five and four target cases, respectively.

Two additional subplots are added to each figure. The bottom left subplot is designed to show the pixel pair responsible for the worst-case slope in the five pixel pair case. Although the colorbar to the right of this subplot shows a graduated scale, the values displayed in the subplot are all integer values. The subplot in the lower right of each figure is the standard deviation of horizontal position, σ_h based on $(\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})^{-1}$, with $\mathbf{R} = \sigma^2\mathbf{I}$, such that $\sigma_h = \sigma\sqrt{(\mathbf{H}^T\mathbf{H})^{-1}_{11} + (\mathbf{H}^T\mathbf{H})^{-1}_{22}}$, and $\sigma = 1$ in the cases shown.

Specifically, Figures 5.26 through 5.29 show the position and position plus attitude cases for the orthogonal configurations $1a$ and $1b$, respectively. The symmetry of the pixel locations produces symmetry in all of the corresponding plots. Plots of $HPE_{_{TD}}$ in the upper two subplots of the position only cases show that the $HPE_{_{TD}}$ is more effectively reduced going from the four pixel pair case to the five pixel pair case, in general, when the added pair falls in the regions either inside all of the baseline pixel pairs or when it falls on a line originating from the center of the pixel plane and radiating out between pixel pairs. The "influence" subplot in the lower left hand corner, indicating the pixel pair with the greatest influence on the worst-case slope at each step, more clearly shows that this line is the transition region between between influence regions established by the individual pixel pairs.

The regions exhibiting the worst $HPE_{_{TD}}$ values in the top two subplots estimating position only tend to fall on lines that radiate from the center out through the baseline pixel locations, showing that added pixel pairs outside of the baseline locations that share an x or y coordinate value (or approximate value) do not improve the integrity. The "normalized" subplot in the upper right corner in fact shows that there is a potential to make the $HPE_{_{TD}}$ value worse by adding another image-based measurement. This development seems counter-intuitive, and as a result is investigated in greater detail later in this chapter using a GPS analogy (see Section 2.5). The standard deviation of the position only estimation cases falls below the $HPE_{_{TD}}$ values in all cases. Recall that the $HPE_{_{TD}}$ metric is also a square-rooted term based on Equation (4.18).

Continuing with Figures 5.26 through 5.29, the estimation using both position and attitude also display excellent symmetry, which is reasonable given the symmetric distribution of the pixel pairs. However, the nature of the influence pattern of the individual pixel locations is clearly different from the position only case. The magnitude of the $HPE_{_{TD}}$ metric has increased across all plots, as expected, but there is also a high degree of variability in both the $HPE_{_{TD}}$ term and the regions of influence for a given pixel pair. This variability is potentially tied to a phasing phenomenon in the new slope computation, where with the addition of the attitude states to the $\mathbf{H}$ matrix, the sinusoidal representations of the new slope calculation, as exemplified in Figure 5.3 for the estimating position only case, now become closer to in-phase for all of the pixel pairs. In the symmetric cases discussed, the amplitudes for the new slope computations are also closer in relative value. These facts, coupled with a sizeable change in relative scale between the $\mathbf{H}$ matrix elements related to the position states and those related to the attitude states, serve to increase numerical sensitivity to small changes in the attitude formulations, meaning that minor shifts in the angular error can cause exaggerated growth in the $HPE_{_{TD}}$ term. This concept is probably best seen in the lower left subplot of Figure 5.29, where the target that has the worst-case slope varies significantly over the different geometries represented by changing the location of the fifth target.

Similar observations can be made in the asymmetric cases shown if Figures 5.30 through 5.33. However, in these cases there is more often significant improvement in the integrity metric, $HPE_{_{TD}}$, between the baseline four pixel pair case and five pixel pair case as most clearly shown in the "normalized" subplots in the upper right corner of each figure. In comparison with the symmetric cases, where the best-case improvement was on the order of $20-30$ percent, the improvement seen in these asymmetric cases goes as high as 80 percent in the estimating position plus attitude cases.

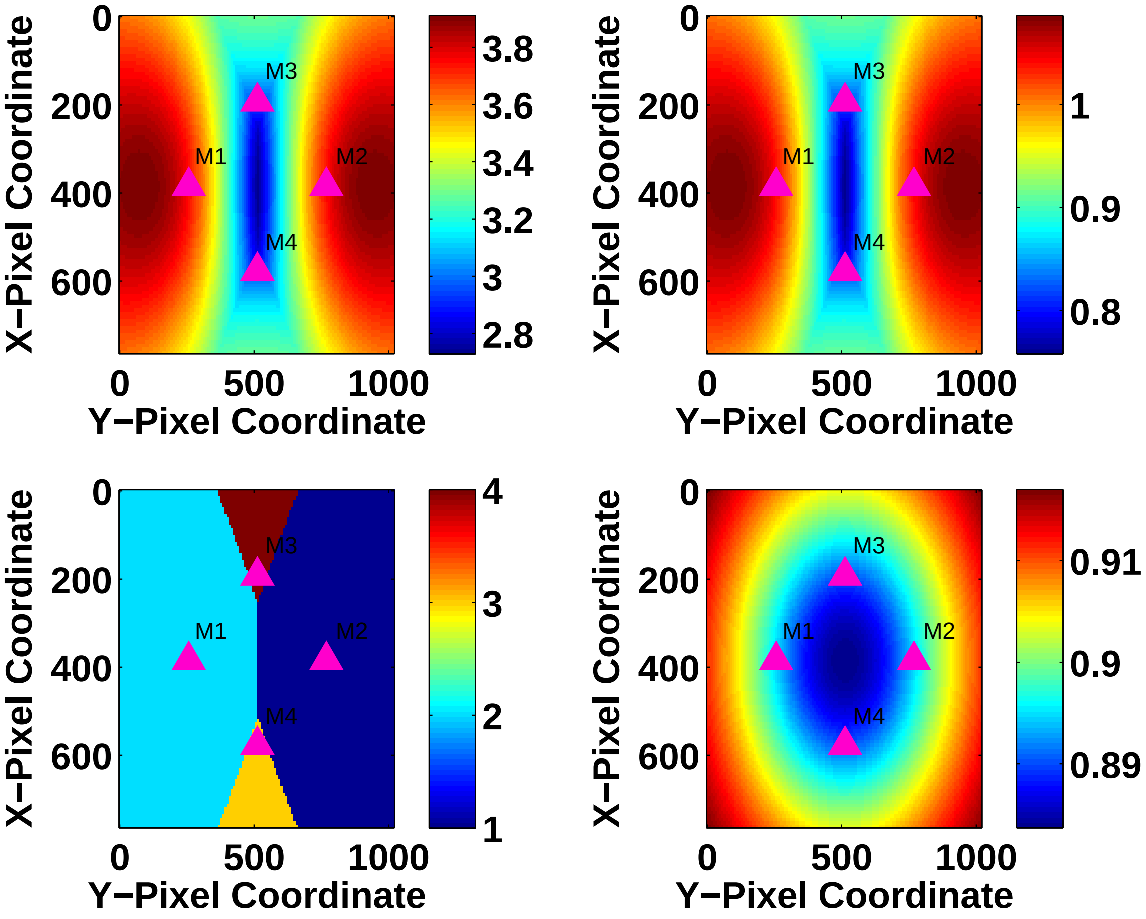

Figure 5.26: Impact of Adding New Measurement in Estimating Position Only Case, Symmetric Geometry $(1a)$. *Top Left*: HPE_{TD} as function of new pixel pair added to baseline four pixel pair set; *Top Right*: HPE_{TD} as function of new pixel pair added to baseline four pixel pair set, normalized by the maximum new slope in the four pixel pair case multiplied by the new TD value; *Bottom Left*: Regions of pixel pair influence; *Bottom Right*: Standard deviation of the horizontal position error (in meters).

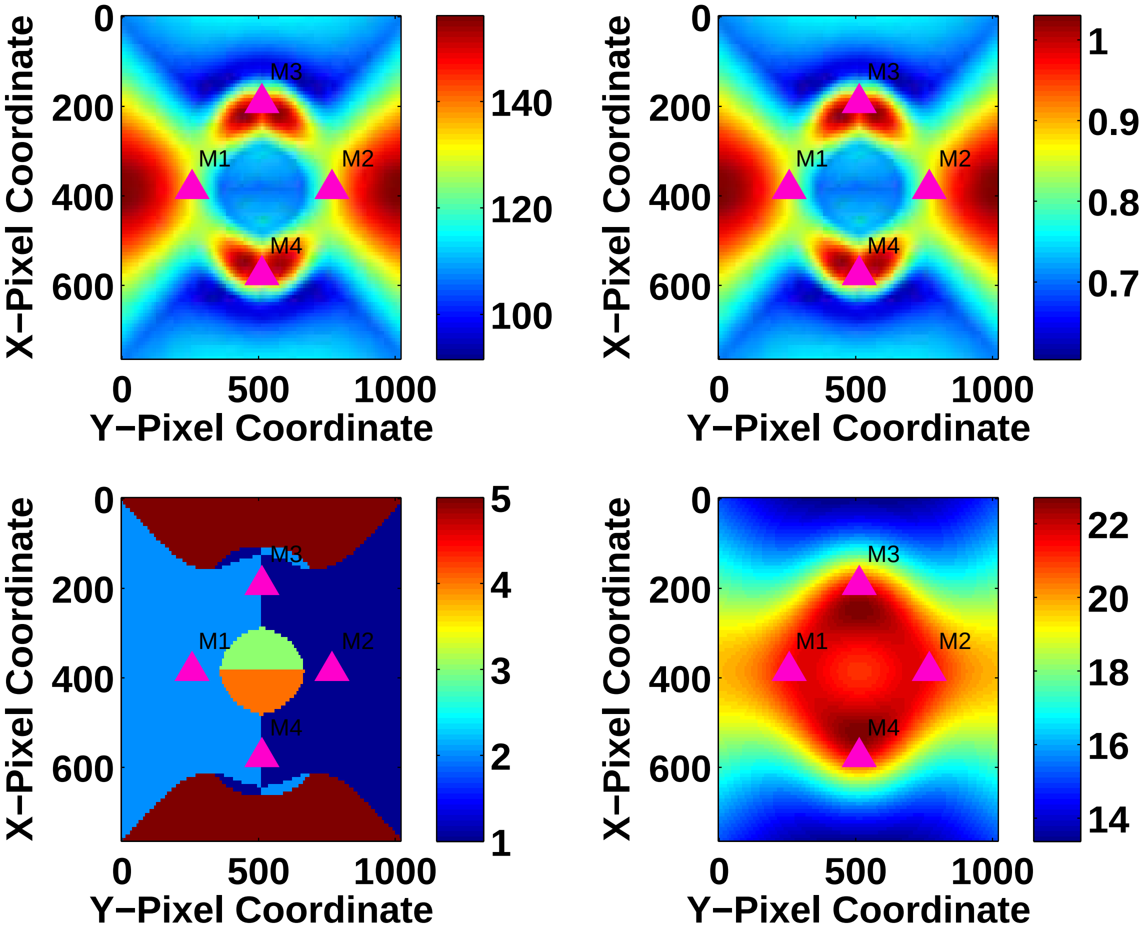

Figure 5.27: Impact of Adding New Measurement in Estimating Position and Attitude Case, Symmetric Geometry $(1a)$. *Top Left*: HPE_{TD} as function of new pixel pair added to baseline four pixel pair set; *Top Right*: HPE_{TD} as function of new pixel pair added to baseline four pixel pair set, normalized by the maximum new slope in the four pixel pair case multiplied by the new TD value; *Bottom Left*: Regions of pixel pair influence; *Bottom Right*: Standard deviation of the horizontal position error (in meters).

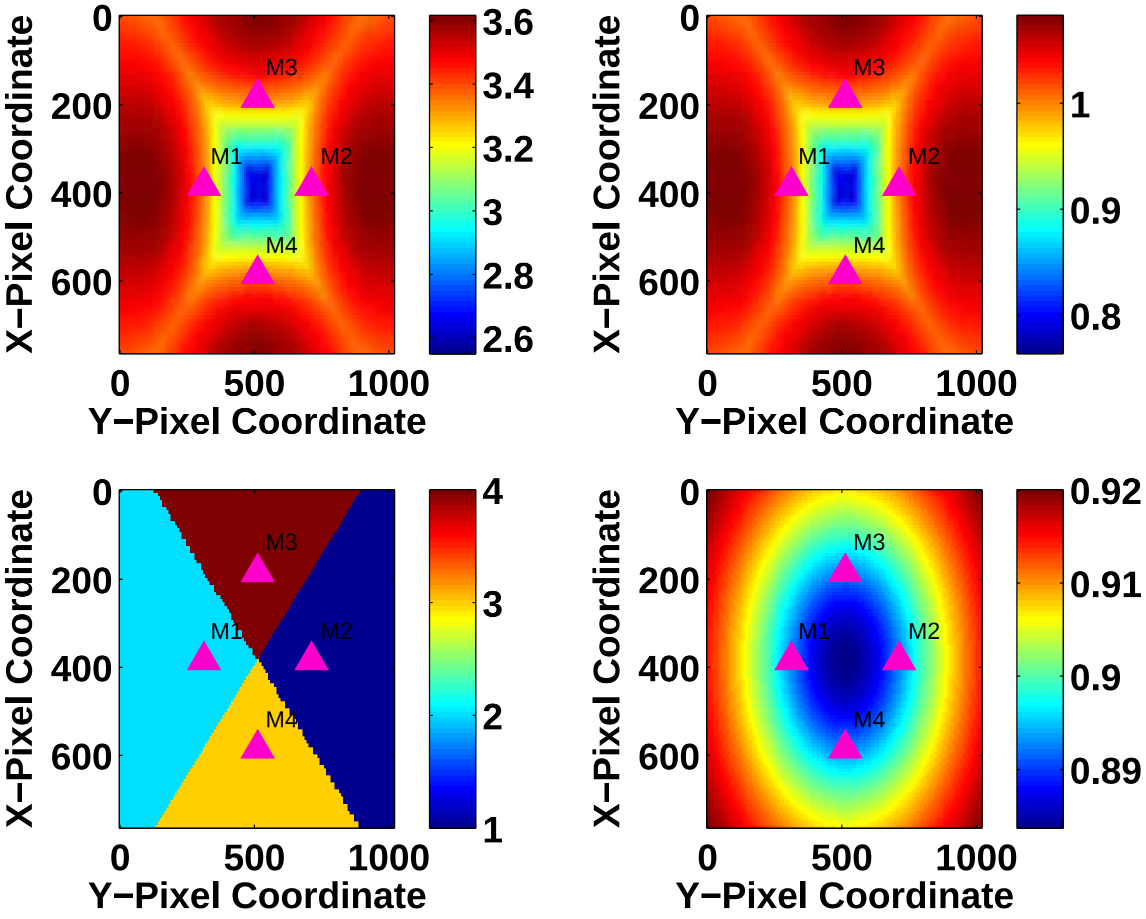

Figure 5.28: Impact of Adding New Measurement in Estimating Position Only Case, Symmetric Geometry ($1b$). *Top Left*: HPE_{TD} as function of new pixel pair added to baseline four pixel pair set; *Top Right*: HPE_{TD} as function of new pixel pair added to baseline four pixel pair set, normalized by the maximum new slope in the four pixel pair case multiplied by the new TD value; *Bottom Left*: Regions of pixel pair influence; *Bottom Right*: Standard deviation of the horizontal position error (in meters).

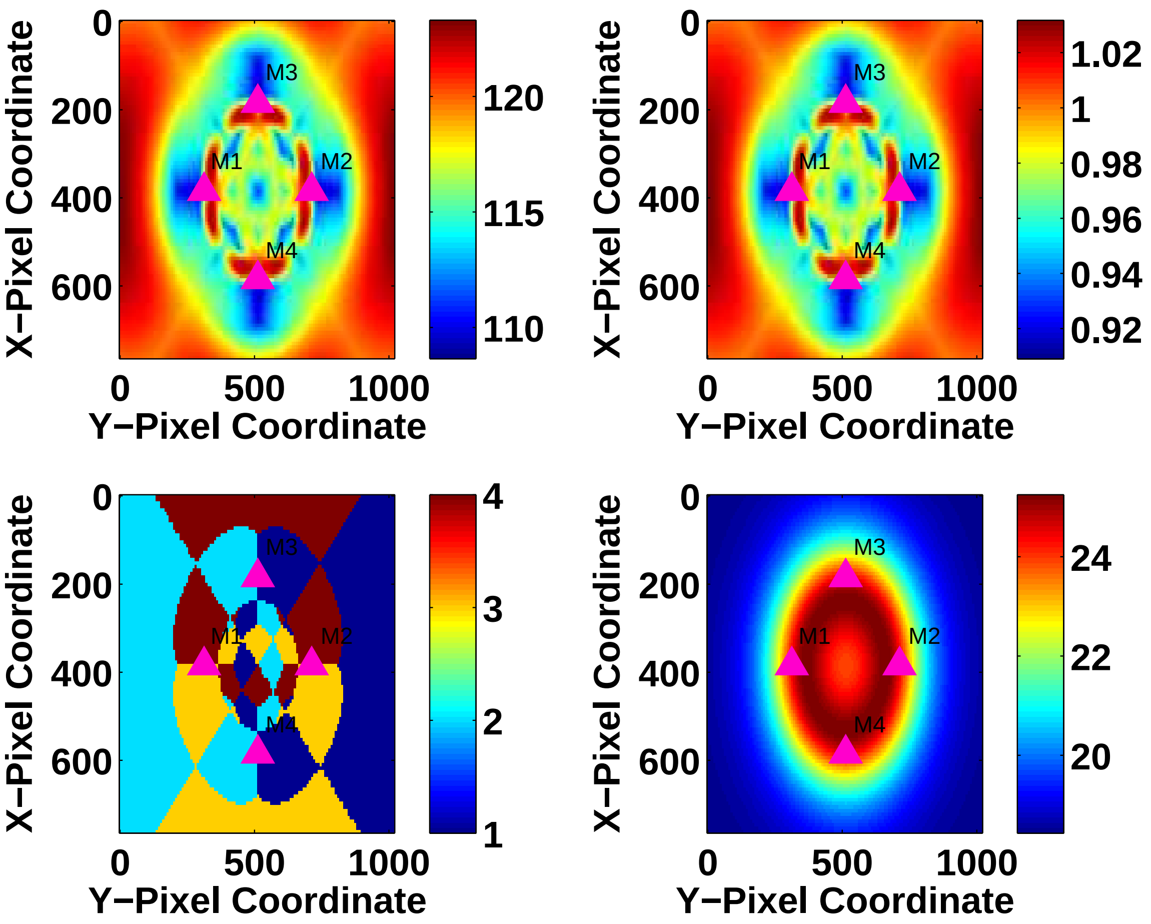

Figure 5.29: Impact of Adding New Measurement in Estimating Position and Attitude Case, Symmetric Geometry $(1b)$. *Top Left*: HPE_{TD} as function of new pixel pair added to baseline four pixel pair set; *Top Right*: HPE_{TD} as function of new pixel pair added to baseline four pixel pair set, normalized by the maximum new slope in the four pixel pair case multiplied by the new TD value; *Bottom Left*: Regions of pixel pair influence; *Bottom Right*: Standard deviation of the horizontal position error (in meters).

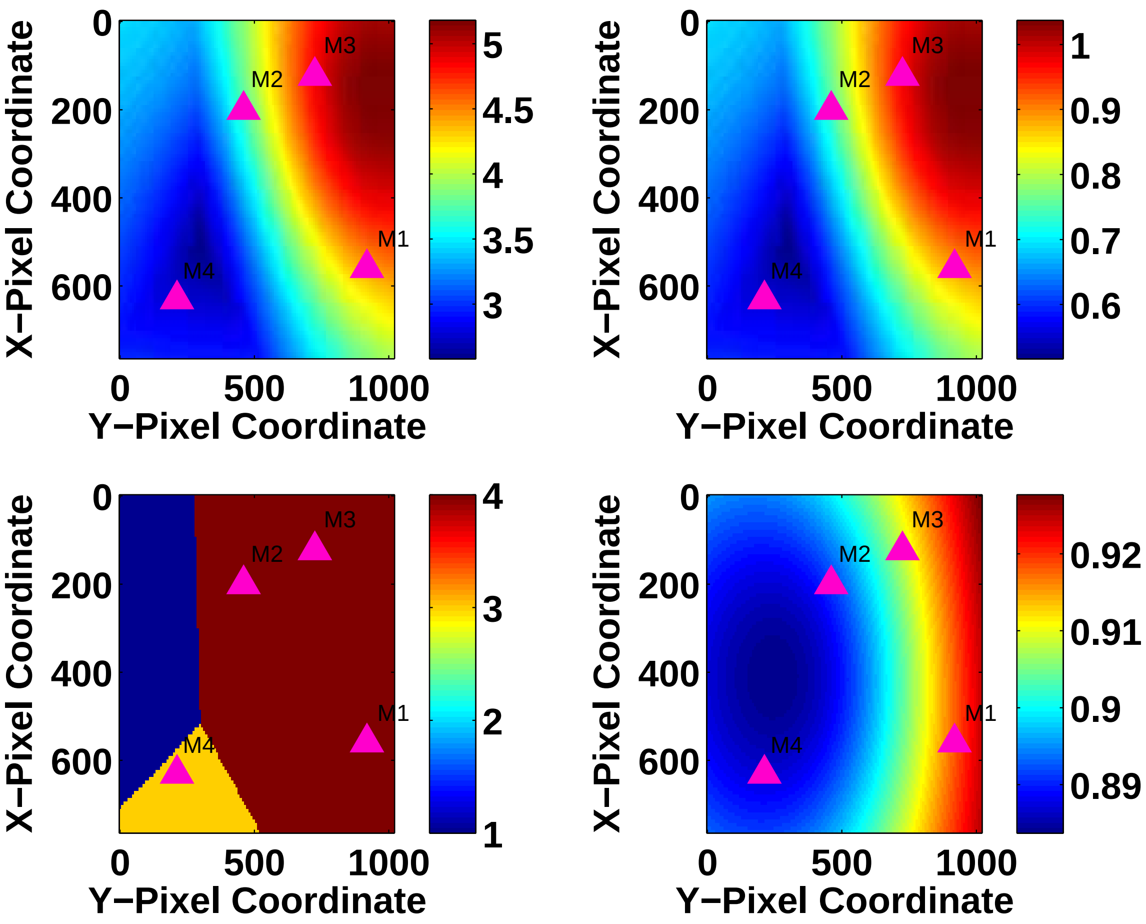

Figure 5.30: Impact of Adding New Measurement in Estimating Position Only Case, Asymmetric Geometry $(2a)$. *Top Left*: HPE_{TD} as function of new pixel pair added to baseline four pixel pair set; *Top Right*: HPE_{TD} as function of new pixel pair added to baseline four pixel pair set, normalized by the maximum new slope in the four pixel pair case multiplied by the new TD value; *Bottom Left*: Regions of pixel pair influence; *Bottom Right*: Standard deviation of the horizontal position error (in meters).

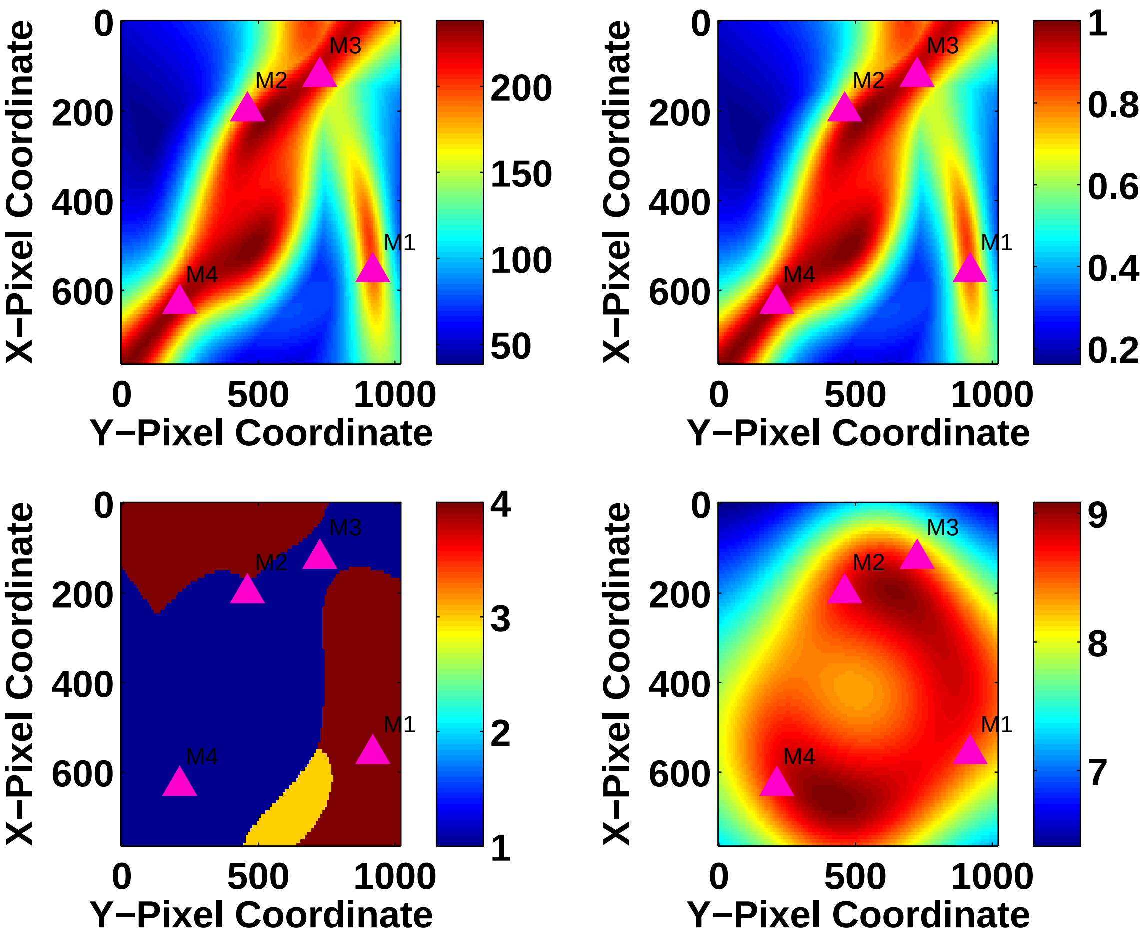

Figure 5.31: Impact of Adding New Measurement in Estimating Position and Attitude Case, Asymmetric Geometry $(2a)$. *Top Left*: HPE_{TD} as function of new pixel pair added to baseline four pixel pair set; *Top Right*: HPE_{TD} as function of new pixel pair added to baseline four pixel pair set, normalized by the maximum new slope in the four pixel pair case multiplied by the new TD value; *Bottom Left*: Regions of pixel pair influence; *Bottom Right*: Standard deviation of the horizontal position error (in meters).

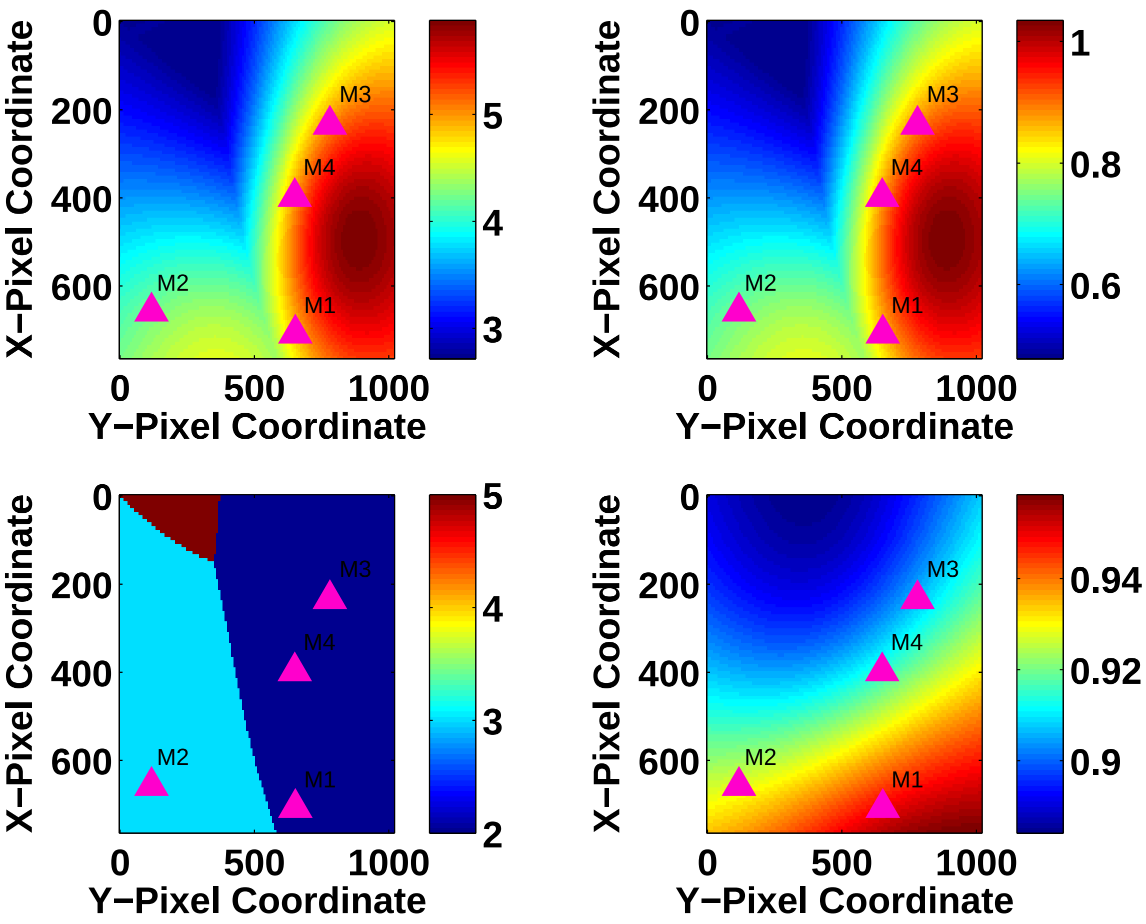

Figure 5.32: Impact of Adding New Measurement in Estimating Position Only Case, Second Asymmetric Geometry $(2b)$ Example. *Top Left*: HPE_{TD} as function of new pixel pair added to baseline four pixel pair set; *Top Right*: HPE_{TD} as function of new pixel pair added to baseline four pixel pair set, normalized by the maximum new slope in the four pixel pair case multiplied by the new TD value; *Bottom Left*: Regions of pixel pair influence; *Bottom Right*: Standard deviation of the horizontal position error (in meters).

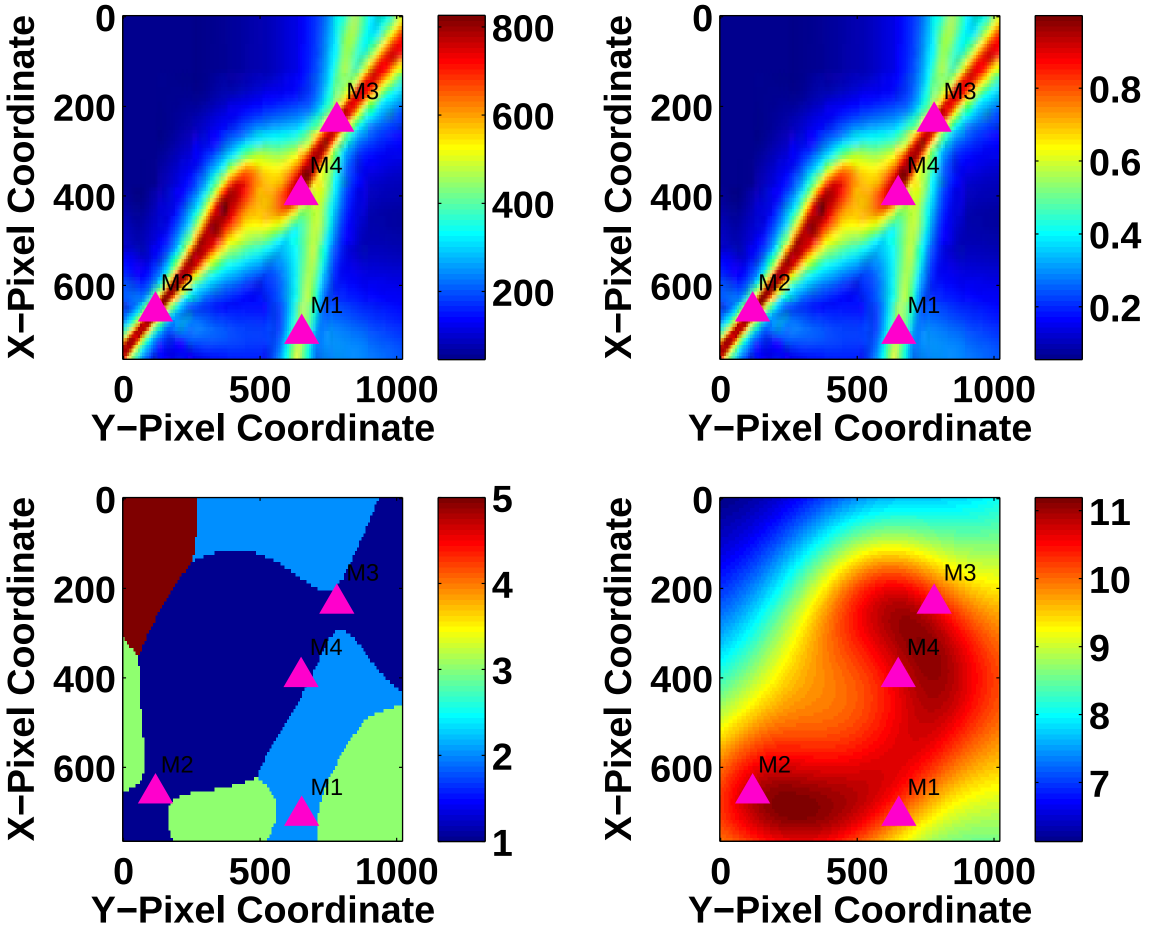

Figure 5.33: Impact of Adding New Measurement in Estimating Position and Attitude Case, Second Asymmetric Geometry ($2b$) Example. *Top Left*: HPE_{TD} as function of new pixel pair added to baseline four pixel pair set; *Top Right*: HPE_{TD} as function of new pixel pair added to baseline four pixel pair set, normalized by the maximum new slope in the four pixel pair case multiplied by the new TD value; *Bottom Left*: Regions of pixel pair influence; *Bottom Right*: Standard deviation of the horizontal position error (in meters).

5.5 *Investigation of Failure to Improve Integrity by Increasing Redundancy*

As shown earlier, when changing from a baseline four pixel pairs to a new five pixel pair case in the image-based scenario, the $HPE_{_{TD}}$ plotted over varying possible locations for the added pixel pair showed that the addition of another pixel pair (increasing overall measurement redundancy) was not guaranteed to improve the worst case value of $HPE_{_{TD}}$ over the lower measurement case. This result seemed counterintuitive and was not perfectly anticipated. In order to draw a comparison to satisfy the question as to whether it could be possible to indeed worsen an expected position error bound by increasing the measurement redundancy, analysis was done using the GPS RAIM methodology, which is more mature in its development than image-based integrity.

Recall that the $HPE_{_{TD}}$ function uses the worst case slope multiplied by a bias term set to be equivalent to the chi-square threshold value computed using a fixed probability of false alarm (p_{fa}) and the number of degrees of freedom (dof) in the formulation (e.g., for the GPS case the dof is given as $m - n$ and for the image-based case the dof is given by $2m - n$, where m is the number of targets and n is the number of states).

It is important to note that, when the number of targets is increased, the number of degrees increases resulting in a different threshold value. For direct comparison between cases involving a different number of targets, the largest threshold value is used in both cases to provide additional commonality between the cases and limit the difference in $HPE_{_{TD}}$ to only a function of the worst case slope. To begin, the first observation matrix chosen for this operation was the well established six GPS pseudorange measurement set published in [18]:

$$\mathbf{H} = \begin{bmatrix} 0.7460266527 & -0.4689257437 & 0.4728137904 & 1 \\ -0.8607445743 & -0.3446039300 & 0.3746557209 & 1 \\ 0.2109370676 & 0.3502943374 & 0.9125784518 & 1 \\ -0.0619331310 & -0.49677359072 & 0.8656891623 & 1 \\ -0.7248969588 & 0.4759681238 & 0.4979746422 & 1 \\ -0.4009266538 & 0.1274180997 & 0.9072058455 & 1 \end{bmatrix} \tag{5.4}$$

with the subsequent slope calculation producing:

$$\mathbf{Slopes} = [2.14370 \;\; 1.09937 \;\; 0.91676 \;\; 1.22794 \;\; 1.19918 \;\; 0.27536] \tag{5.5}$$

The resultant $HPE_{_{TD}}$ is found to be 11.70785 using the $m - n = 6 - 4 = 2$ degrees of freedom. Using the maximum slope for the six measurement case shown, but increasing the number of degrees of freedom to 3, subsequently changing the threshold value, increases the $HPE_{_{TD}}$ to 12.30139. For comparison, by introducing a new row into the observation matrix given in Equation (5.4), with the values of the new row chosen to be the same as those seen in row two of Equation (5.4), the slopes change to the following:

$$\mathbf{Slopes} = [2.16114 \;\; 0.43942 \;\; 0.72751 \;\; 1.10406 \;\; 1.22625 \;\; 0.25898 \;\; 0.43942] \tag{5.6}$$

with the $HPE_{_{TD}}$ value calculated for 3 degrees of freedom equal to 12.40145. This means that for the same bias level, which is chosen to be threshold value for the higher number of measurements in these scenarios, the worst case $HPE_{_{TD}}$ actually increases slightly when adding an additional measurement in the specific geometry shown. The ratio of the change is a modest 1.008, but this change is on the order observed in the image-based case, where normalization by the lower number of measurements values, adjusted for degrees of freedom, were in the range 1 - 1.03. It should also be noted that in the image-based case, the dof increases by two due to the measurement pair structure.

Recall that this study was intended to demonstrate that the non-intuitive result is possible, even in a well-known case, removing concern that the non-intuitive result is incorrect in the image-based case. This required at least one specific example where an increase in redundancy did not serve to reduce the HPE_{TD} level when compared to a lower number measurement case. The observation matrix for this baseline example was selected because it is well published and easily verifiable. The fact that two GPS satellites would not actually appear in the same exact position was put aside to show the influence of the mathematics. As another example, a different observation matrix was generated, using newer GPS data from GPS week 1408 relative to a user position near Palmdale, California, inspired by a scenario seen in [119]. Selecting five GPS measurement observations from this scenario, the generated $\mathbf{H}$ matrix is:

$$\mathbf{H} = \begin{bmatrix} 0.420274479662478 & -0.240841355313369 & 0.874851303545496 & 1 \\ 0.299507979739954 & 0.375272152076612 & 0.877191986937797 & 1 \\ -0.419134304977878 & 0.217386297643779 & 0.881515531336483 & 1 \\ 0.613074918463151 & 0.673941277080147 & 0.41224058436668 & 1 \\ -0.127580307459073 & -0.784866372842578 & 0.606389348463166 & 1 \end{bmatrix} \tag{5.7}$$

with the resulting slope values

$$\mathbf{Slopes} = [1.83485 \;\; 0.50424 \;\; 3.45636 \;\; 2.40915 \;\; 2.41307] \tag{5.8}$$

An additional satellite is actually available and, for comparison, added to the observation matrix in the last row:

$$\mathbf{H} = \begin{bmatrix} 0.420274479662478 & -0.240841355313369 & 0.874851303545496 & 1 \\ 0.299507979739954 & 0.375272152076612 & 0.877191986937797 & 1 \\ -0.419134304977878 & 0.217386297643779 & 0.881515531336483 & 1 \\ 0.613074918463151 & 0.673941277080147 & 0.41224058436668 & 1 \\ -0.127580307459073 & -0.784866372842578 & 0.606389348463166 & 1 \\ 0.277903971977838 & -0.0720360230624255 & 0.95790406291042 & 1 \end{bmatrix} \tag{5.9}$$

with the slope values changing as follows:

$$\mathbf{Slopes} = [0.98560 \;\; 0.36975 \;\; 3.48776 \;\; 1.91840 \;\; 2.41001 \;\; 0.48353] \tag{5.10}$$

When comparing Equation (5.8) and Equation (5.10) it is clear that the dominant slope is still the number three element, but the value has increased slightly while most other elements have decreased or remained nearly unchanged. Using the same bias level to compare HPE_{TD} between the five and six measurement sets in this scenario, the value of HPE_{TD} again increases with increase in redundancy, changing from 18.87692 to 19.04842 for this specific examination. The ratio of change between the six measurement and five measurement cases, respectively, is 1.009, again consistent with the range observed in the image-based case. The result is increased confidence, that under the conditions posed in the investigation of geometry in the image-based measurements, it is possible that adding additional measurements, thereby increasing redundancy, does not guarantee an improvement of HPE_{TD} over all possible geometries. This consequence also reinforces the importance of geometry screening, where possible, in both GPS and image-based navigation integrity methods.

5.6 *Investigation of Linearization*

An additional study was done in this dissertation to examine the behavior of the $HPE_{_{TD}}$ metric under potential linearization error with respect to estimating position only. Simulations were done by first using a baseline geometry using different subsets from Table 5.1 to build truth target locations relative to a known vehicle position in the NED reference frame. Then errors in vehicle position were sequentially introduced in first the North and then the East coordinates of the vehicle while the truth targets remained unchanged. The $\mathbf{H}$ matrix was recomputed at each instance of error and used to formulate a new $HPE_{_{TD}}$ value for each point. Figures 5.34 through 5.37 show the results for four selected cases, $1a$,$1b$,$2a$, and $2c$, which are two symmetric and two asymmetric examples.

In each of the figures there are two columns of subplots, reflecting the results for two different altitudes (or ranges to a target plane). The column on the right is for a distance of $1000m$, while the column on the left is for a distance of $15m$, with the shorter distance chosen to be more likely encountered in the air-to-air refueling scenario discussed earlier in the chapter. A baseline $HPE_{_{TD}}$ value based on the true $\mathbf{H}$ matrix is plotted in conjunction with the $HPE_{_{TD}}$ resulting from the linearization error in a given direction, identified on the x-axis of the plots. Vertical lines reflecting the $\pm 6\sigma$ thresholds are also shown.

Values taken from the simulations are summarized in Table 5.2, with N and E denoting the North and East directions, respectively. The $||\Delta HPE_{_{TD}}||$ term used in the last column describes the absolute value of the distance between the baseline $HPE_{_{TD}}$ and the $HPE_{_{TD}}$ resulting from error, as measured at the $\pm 6\sigma$ thresholds in North and East (N/E). The value of $||\Delta HPE_{_{TD}}||$ was found to be equivalent on either side of the zero point in each directional test for each case. It should be noted that the values for the $1000m$ distance are recorded in meters while those for the $15m$ distance are recorded in centimeters.

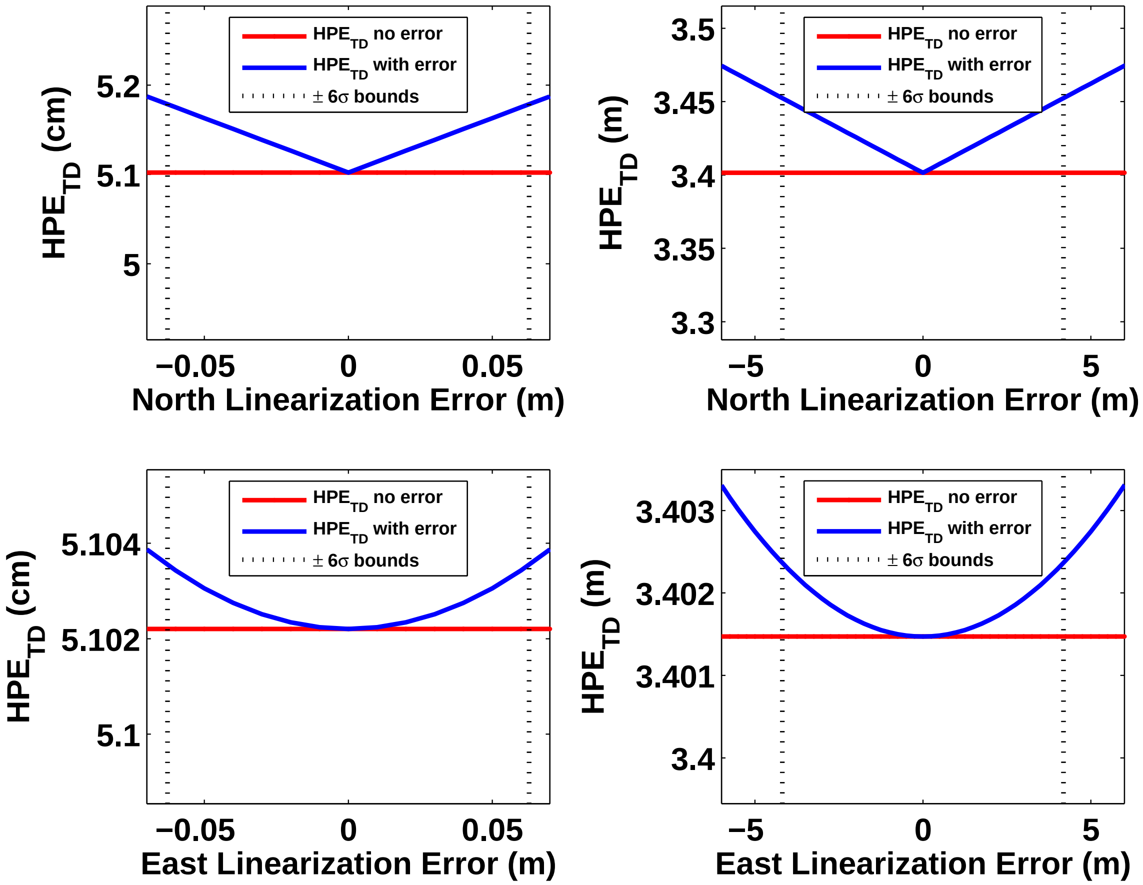

Figure 5.34: HPE_{TD} versus Linearization Error for Symmetric Geometry $(1a)$. Left column of subplots reflect $15m$ distance while right column of subplots show results for $1000m$ (Note difference in scales between left and right columns).

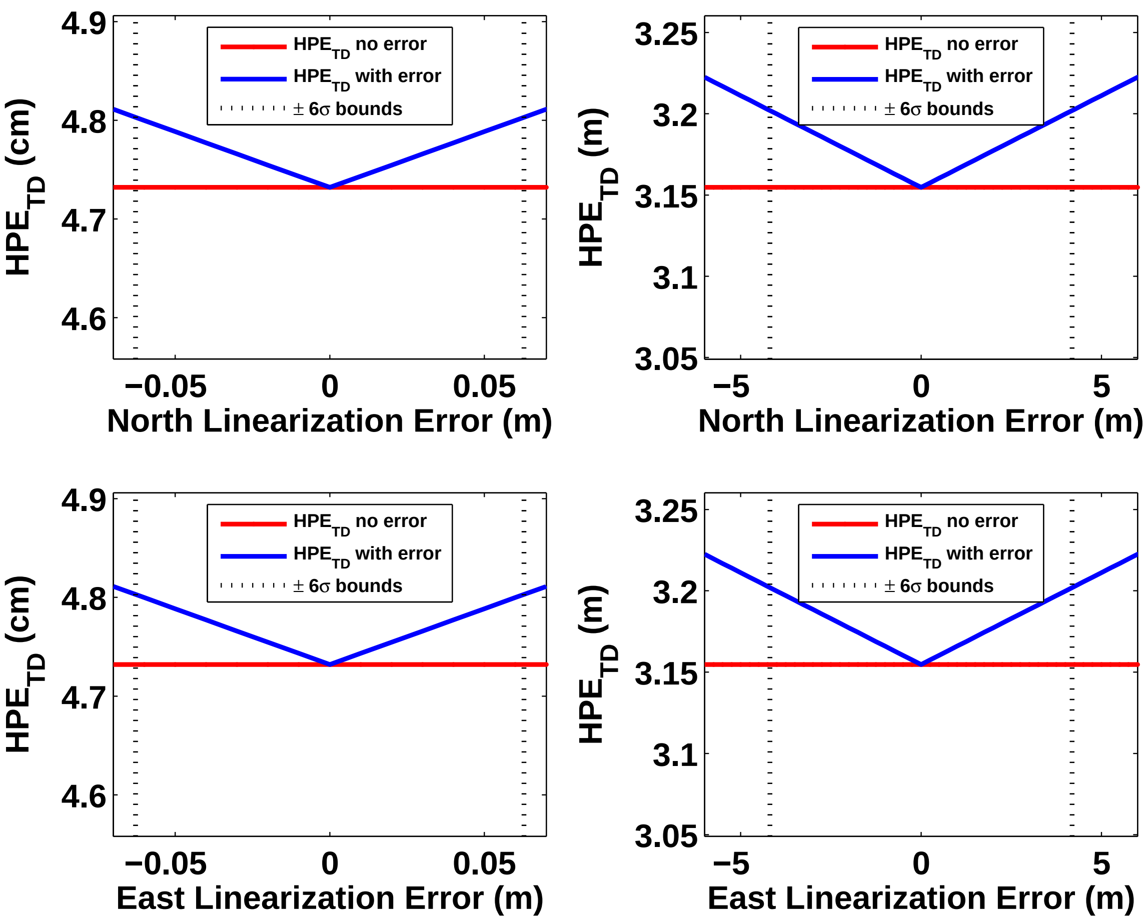

Figure 5.35: HPE_{TD} versus Linearization Error for Symmetric Geometry ($1b$). Left column of subplots reflect $15m$ distance while right column of subplots show results for $1000m$ (Note difference in scales between left and right columns).

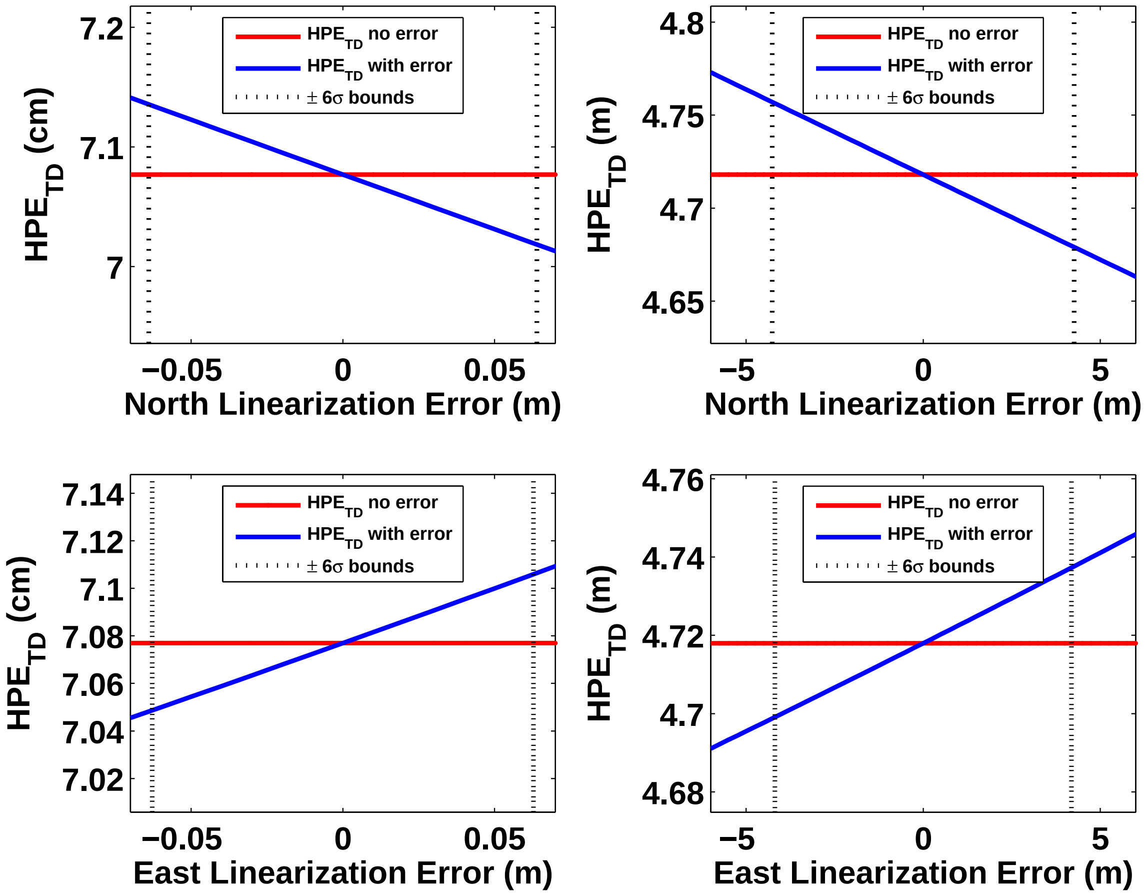

Figure 5.36: HPE_{TD} versus Linearization Error for Asymmetric Geometry ($2a$). Left column of subplots reflect $15m$ distance while right column of subplots show results for $1000m$ (Note difference in scales between left and right columns).

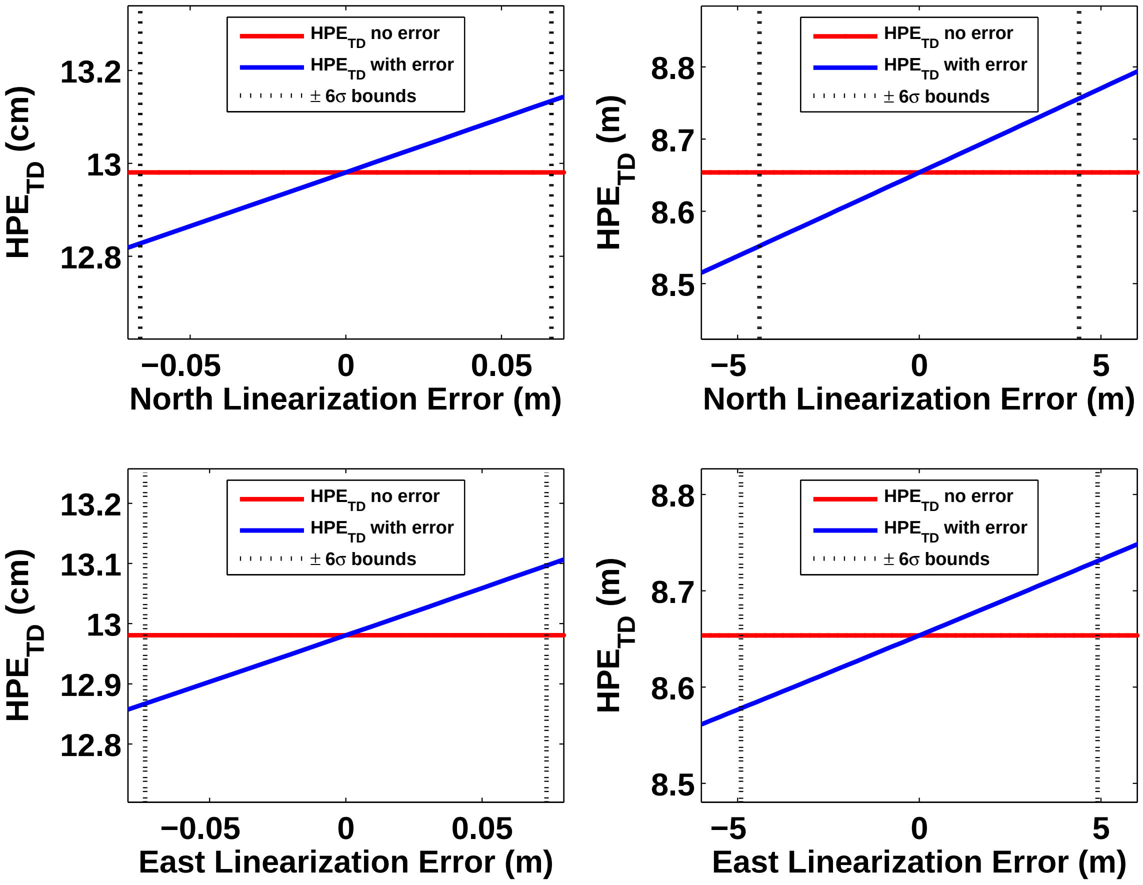

Figure 5.37: HPE_{TD} versus Linearization Error for Asymmetric Geometry $(2c)$. Left column of subplots reflect $15m$ distance while right column of subplots show results for $1000m$ (Note difference in scales between left and right columns).

Table 5.2: Linearization Summary; Estimating Position Only

Geometry Identifier	Range to Target	1σ Value (σ_N/σ_E)	Baseline (no error) HPE_{TD}	$\|\|\Delta HPE_{TD}\|\|$ between no error and error cases at $\pm 6\sigma$ (N/E)
$1a$	$1000m$ $15m$	$0.70m/0.70m$ $1.05cm/1.05cm$	$3.40m$ $5.10cm$	$0.05m/0.001m$ $0.07cm/0.02cm$
$1b$	$1000m$ $15m$	$0.70m/0.70m$ $1.05cm/1.05cm$	$3.16m$ $4.73cm$	$0.05m/0.05m$ $0.07cm/0.07cm$
$2a$	$1000m$ $15m$	$0.71m/0.70m$ $1.07cm/1.05cm$	$4.72m$ $7.08cm$	$0.04m/0.02m$ $0.06cm/0.03cm$
$2c$	$1000m$ $15m$	$0.73m/0.82m$ $1.1cm/1.2cm$	$8.65m$ $12.98cm$	$0.10m/0.08m$ $0.15cm/0.11cm$

When evaluating the plots it is clear to see that in nearly all instances the HPE_{TD} error-based line is nearly perfectly linear between the thresholds, with one exception. In the symmetric geometry case, $1a$, of Figure 5.34 the plots of the East directional error appear parabolic. Since the scale of the change is very small across the region of interest (on the order of fractions of millimeters), the change in HPE_{TD} can be still be considered reasonably linear. What is interesting to note is that when comparing this figure of case $1a$ with Figure 5.35 for case $1b$, the parabolic shape can be attributed to the fact that the pixel locations are not equidistant in the former case. This demonstration mimics that seen earlier in the geometry study with Figures 5.26 and 5.28 where the smallest values of HPE_{TD} fall in a rectangular region in the first plot and a nearly perfect square in the second. The directions have changed in the linearization plots due to coordinate transformations.

Comparisons between results at the two different ranges show a consistent scaling across the different parameters. For example, when taking the ratios of the σ values, the ratios of the HPE_{TD} baselines, or ratios of the $||\Delta HPE_{TD}||$ between the $1000m$ and 15 meter ranges, the results match the ratio of the distances themselves, $1000/15$. This is consistent with the observations made earlier when comparing HPE_{TD} versus altitude for the different cases. It is suspected that this would be true in most cases under the present assumptions, since the worst-case angular error is amplified by a particular geometry, and if the geometry is consistent when changing the range parameter, the worst case new slope should also stay consistent.

In general, there do not appear to be significant problems with linearization inside the regions of interest in this data. Several factors may help account for this fact. First, the analysis is done under the assumption that the target locations are known. In the study by [119], errors resulting from target estimation were thought to negatively impact linearization. Secondly, this investigation is using a least-squares method on an epoch-by-epoch basis, avoiding potential accumulation of errors in an Extended Kalman Filter. Recently uncovered work by [52] attempts to address such errors with the Extended Kalman Filter by evaluating the $\mathbf{H}$ matrix using the target estimate from the first time it is detected, sacrificing accuracy in the observation matrix for stability of the system.

Initial investigation was also done regarding linearization when estimating both position and attitude. However, comprehensive results were not achievable, and are left to future work. What is known, or at least suspected, is that inclusion of attitude should create heightened sensitivity due to the attitude errors influencing the DCM relating the body and navigation frame, which impacts both position and attitude elements in the $\mathbf{H}$ matrix.

The next chapter provides a summary and additional conclusions, as well as identifying the contributions to the state of science accomplished through this research.

VI. Conclusions

This dissertation develops an initial framework for image-based integrity, extending GNSS integrity concepts to an image-based navigation system dependent on an entirely different measurement structure. This chapter first presents conclusions and discussions regarding the total research effort. Potential areas of future related research are then described before a final brief summary is given.

6.1 Conclusions

This research effort has established an initial framework for image-based navigation integrity through a series of significant steps described in the following paragraphs. The work presented in this dissertation is the first known investigation into image-based integrity, and was accomplished with the purpose of establishing a performance baseline for image-based navigation integrity and presenting the development and analysis in terms that could be easily understood by the navigation community. An assured navigation solution is of critical importance in navigation systems supporting safety-of-flight and/or safety-of-life operations, and thus, as image-based navigation becomes a viable augmentation or alternative to GNSS navigation, it is necessary to develop this baseline now. As mentioned in earlier chapters, GNSS integrity has been the subject of extensive research for more than two decades and even now continues to evolve. A significant goal of this dissertation was to initiate a similar magnitude research vein for image-based navigation systems.

A key contribution of this research is in the rigorous development of a mathematical relationship relating the horizontal position error, which is not directly observable, to an associated test statistic that is outwardly observable, in the bearing-type measurement case of the image-based navigation systems described in this document. The GNSS community developed a core relationship, focusing primarily on single-failure or uncorrelated two-failure type conditions of pseudorange measurements. This research extends these concepts in developing an approach to address the impact of angular errors influencing the image-based measurements composed of paired elements potentially having both elements corrupted by such error.

The algorithm developed and evaluated in this research is shown to handle cases where a single element of the pixel pair is affected by a bias term, paralleling the result of proposed GNSS RAIM algorithms when treating each element as an individual measurement. However, where the basic GNSS RAIM algorithm fails to compensate for an additional adjacent element of the image-based measurement containing a bias, the newly developed algorithm produces a value indicative of the horizontal position error and test statistic relationship allowing prediction of the worst case position error, in a deterministic evaluation, relative to a specified threshold defined by user probability of false alarm requirements. In addition, GNSS concepts for handling dual biases appear to be conservative, and not ever observed to be more precise in this research, than the framework described in this research effort, which accurately predicts the “worst-case” result while improving availability, in the deterministic evaluations, over the former conservative approach.

Another contribution in this research is the evaluation of the impact of attitude on the image-based integrity result. The results show that when extending the image-based navigation problem from one where only position estimation is done to one where both position and attitude are being estimated, there is a significant increase in the magnitude of the potential horizontal position error before possible detection at a defined threshold could occur. This heightened uncertainty created through the required augmentation in the observation matrix, combined with the fact that the image-based errors are expected to exhibit angular error characteristics, create a high degree of sensitivity in the integrity model evaluated here. This drives the assertion that integrity in image-based navigation is significantly benefitted when mechanisms for attaining high accuracy attitude information are employed (e.g., use of high-quality IMUs).

Additional contributions seen in this research include the study and evaluation of the relationship between the image-based integrity performance metric and altitude, the investigation and evaluation of the impact of geometry on image-based navigation integrity, and examination of linearization when estimating position only, with respect to the image-based integrity performance metric. In evaluating image-based integrity over multiple altitude scenarios, a linear growth pattern was observed, one that enhances the ability to make coarse predictions regarding potential integrity limits, potentially before a specific pixel pair geometry is known. Recall that unlike in a GNSS scenario, where the satellite orbits are known and navigation integrity based on known geometries can be predicted, the image-based measurement geometries are not necessarily predictable, requiring an epoch-by-epoch approach. Thus, any *a priori* information available to the user in an image-based scenario, like altitude, may enhance integrity.

Specifically focusing on geometry, the evaluation of large numbers of random geometries in scenarios based on varying numbers of pixel pairs, show that similar empirical distributions start to emerge allowing, at a minimum, calculations on the order of magnitude of the potential outliers of computed image-based integrity metrics. Additional geometry studies also reveal the impact on integrity when an additional pixel pair is added to the navigation solution. The results prove counterintuitive, in that the additional measurement does not improve integrity in all cases. A detailed comparative evaluation is done in a GNSS scenario proving that, although not as common an occurrence, a similar result is indeed possible. This outcome highlights another difference between integrity for GNSS versus image-based systems. A key result of this analysis on the impact of geometry is in showing that the overall performance of an image-based system is potentially improved by using the integrity framework to focus measurement selection in regions that enhance integrity.

Finally, a brief investigation into the impact of error in the nominal position used for linearization, with respect to the image-based integrity metric shows that in the cases studied, the metric exhibits linear or approximately linear properties across a $\pm 6\sigma$ range of potential position error. Although a similar investigation regarding attitude is not completed, rough approximations describe the possible impact as a function of range from target.

6.2 Future Work

This research effort establishes a rigorous foundation for image-based navigation system integrity. As previously mentioned, the advent of GNSS integrity spawned new research efforts for more than twenty years, and, similarly, there are many opportunities in future work to advance the state-of-the-art presented in this dissertation. To build the initial framework, a number of constraints were put in place that can be eventually loosened as more research is done in this area. These constraints and potential research avenues are described in the following paragraphs detailing future work possibilities.

The research has clearly defined performance in conjunction with geometry screening and evaluation against a threshold based on the probability of false alarm, focusing on the deterministic case assumed to be resultant of the expected value of the measurement model under biased and non-biased measurement conditions. Future work could extend these results to the non-deterministic condition, where an adjustment to the performance metric scale must be considered to account for the distribution of added noise. In the GNSS case this is based on a second integrity parameter, the probability of missed detection, the formulation of which is already described in this dissertation for a chi-square test statistic. Conceptually, this moves to replacement of the TD parameter in the HPE_{TD} metric with a new bias term derived based on the non-deterministic scenario and required probability of missed detection. This step effectively drives the “worst-case” result up the new slope to a place where both the probability of false alarm and probability of missed detection are met simultaneously. Additional future work opportunities exist in incorporating prior failure probabilities (feature matching probabilities), when known, into the equations.

Although a specific statistical distribution is not yet identified to fully characterize the histogram data reflected in this research for random geometries, a number of interesting empirical trends are shown observed in terms of mode and extreme values. These observations are expected to prove useful in the continuing development of an image-based integrity framework. These results serve as a jumping off point for further investigations and statistical analysis in advancing the image-based navigation integrity framework, to include study of non-deterministic scenarios. Characterizing the probability density function of the image-based integrity metric for the random geometries could lead to the ability to predict in advance the integrity limits under anticipated flight profiles.

In order to limit variability while establishing a baseline, the targets observed in an image are assumed to have known locations. This assumption is consistent with the research goal in tailoring toward an air-to-air refueling scenario where points of interest could be known relative to a fixed location in a lead aircraft. However, future work could investigate the impact on image-based integrity when target locations are not definitively known, and have to be estimated. As discussed in the previous chapter, there are potential problems with linearization error when introducing target estimates into the equation, which may be another extension. With regard to linearization error, a full evaluation regarding attitude remains an opportunity for future work as well.

The current research is based on a single pair of biased image-based measurements (extending the single element failure assumption). As the number of failures grow it becomes increasingly difficult to associate error influence to specific measurements. There is an opportunity for future research to extend the single pair failure assumption to a multiple pair failure condition. The later assumption would require specific methods for trying to identify the bad measurement sources, likely requiring decision tree analysis and extensive combinatorial mathematics.

Although this research ultimately focuses an epoch-by-epoch least-squares approach in image-based navigation, earlier chapters in this dissertation reviewed Extended Kalman Filter (EKF) approaches to GNSS/INS integrity. There exists an opportunity for future work in applying concepts developed in this dissertation to a filtered approach and also use resources developed by [119] and modified in this research effort. The Extended Kalman Filter models are practical for the first investigations into an image-based integrity framework because they are consistent with a least-squares approach taken in RAIM processes. However, based on limitations of the EKF described by [119], a future research direction may include the application of image-based integrity in a particle, or equivalent, filter. It must be said that this is anticipated to be a extremely complicated transition, however.

*6.3 **Summary***

The advantages to image-based navigation are many, including improved interference immunity and reduced detectability of the user, in addition to advances in technology making the camera and image-processing equipment potentially very small and less expensive than ever before. Previous research efforts described in earlier chapters show the feasibility of image-based approaches and describe reasonable levels of accuracy, on the order of meters in some cases. However, now that this technology is envisioned for safety critical operations, there is a requirement to assure the correctness of the navigation solution. This research has moved the effort for assured navigation a step forward, but, as described in the previous paragraphs, there is still work to be done.

Bibliography

1. *Federal Radionavigation Plan*. Technical Report DOT-VNTSC-RITA-05-12/DoD-4650.5, Springfield, VA: Joint Publication by US Departments of Defense, Homeland Security, and Transportation, December 2005.

2. Anderson D. A. and Doty J. H. "GPS-IMU-Loran Integration for Airborne Applications." *Proceedings of the National Technical Meeting of The Institute of Navigation*. San Diego, CA, 26-28 January 2004.

3. Anton H. and Rorres C. *Elementary Linear Algebra*. New York, NY: John Wiley & Sons, Inc., 2000.

4. Arthur T. "The Disparity of Parity, Determining Position Confidence Bounds in the Presence of Bias." *Proceedings of the Institute of Navigation GNSS, 18th International Technical Meeting of the Satellite Division*. Long Beach, CA, 13-16 September 2005.

5. Bayoud F. A. "Vision-Aided Inertial Navigation Using Geomatics Approach." *Proceedings of the Institute of Navigation GNSS, 18th International Technical Meeting of the Satellite Division*. Long Beach, CA, 13-16 September 2005.

6. Bhatti U. I., Ochieng W. Y., and Feng S. "Integrity of an Integrated GPS/INS System in the Presence of Slowly Growing Errors. Part I: A Critical Review," *GPS Solutions by Springer*, *II*(3):173–181 (July 2007).

7. Bhatti U. I., Ochieng W. Y., and Feng S. "Integrity of an Integrated GPS/INS System in the Presence of Slowly Growing Errors. Part II: Analysis," *GPS Solutions (Springer-Verlag)*, *II*(3):183–192 (July 2007).

8. Bhatti U. I. "An Improved Sensor Level Integrity Algorithm for GPS/INS Integrated System." *Proceedings of the Institute of Navigation GNSS, 19th International Technical Meeting of the Satellite Division*. Fort Worth, TX, 26-29 September 2005.

9. Bow S.-T. *Pattern Recognition and Image Processing*. New York, New York 10016: Marcel-Dekker, Inc., 1992.

10. Brenner M. "Implementation of a RAIM Monitor in a GPS Receiver and an Integrated GPS/IRS." *Proceedings of ION GPS-90, Satellite Division, Technical Meeting*. September 1990.

11. Brenner M. "Integrated GPS/Inertial Fault Detection Availability." *Proceedings of the ION GPS-95, Satellite Division, Technical Meeting*. Palm Springs, CA, 12-15 September 1995.

12. Brenner M. "Integrated GPS LAAS/Inertial Guidance System Using Multiple Kalman Filters." *Proceedings of the ION GPS-99, Satellite Division, Technical Meeting*. Nashville, TN, 14-17 September 1999.

13. Britting K. R. *Inertial Navigation Systems Analysis*. New York: Wiley-Interscience, 1971.

14. Brock L. D. and Schmidt G. T. Chapter 10 of *Theory and Applications of Kalman Filtering*: "General Questions on Kalman Filtering in Navigation Systems," edited by C.T. Leondes, *NATO AGARDOgraph No. 139*, Technical Editing and Reproduction, Ltd., London, 1970.

15. Brown R. G. "A Baseline GPS RAIM Scheme and a Note on the Equivalence of Three RAIM Methods." *Journal of the Institute of Navigation, Vol. 39, No. 2*. Fall 1992.

16. Brown R. G. "Receiver Autonomous Integrity Monitoring," *Global Positioning System: Theory and Applications, Volume II*, 143–165 (1993).

17. Brown R. G. "Solution of Two-Failure GPS RAIM Problem Under Worst-Case Bias Conditions: Parity Space Approach," *NAVIGATION: Journal of the Institute of Navigation, Vol. 44, No. 4*, 425–431 (1998).

18. Brown R. G. and Chin G. "GPS RAIM: Calculation of Threshold and Protection Radius Using Chi-Square Methods - A Geometric Approach," *Institute of Navigation Special Monograph Series (Red Book), Vol. V*, 155–178 (1998).

19. Brown R. G. and Hwang P. Y. *Introduction to Random Signals and Applied Kalman Filtering*. New York: John Wiley and Sons, 1997.

20. Call C., Ibis M., McDonald J., and Vanderwerf K. "Performance of Honeywell's Inertial/GPS Highbrid (HIGH) for RNP Operations," *Proceedings of the IEEE*, 244–255 (2006).

21. Canny J. F. "A computational approach to edge detection." *IEEE Transactions Pattern Analysis and Machine Intelligence*. 1986.

22. Cheng L., Gong J., Fan C., and Han P. "Robbust Affine Invariant Feature Extraction for Image Matching," *5*(2):246–250 (April 2008).

23. Chin G. Y., Kraemer J. H., and Brown R. G. "GPS RAIM: Screening Out Bad Geometries Under Worst Case Bias Conditions." *Journal of the Institute of Navigation, Vol. 39, No. 4*. Winter 1992-1993.

24. Clot A., Macabiau C., Nikiforov I., and Roturier B. "Sequential RAIM Designed to Detect Combined Step Ramp Pseudo-Range Errors." *Proceedings of the Institute of Navigation GNSS, 19th International Technical Meeting of the Satellite Division*. Fort Worth, TX, 26-29 September 2006.

25. Daly P. and Misra P. N. *GPS and Global Navigation Satellite System (GLONASS)*. Global Positioning System: Theory and Applications Volume II, Washington, D.C.: American Institute of Aeronautics and Astronautics, Inc., 1996.

26. Davison A. "Real-time Simultaneous Localisation and Mapping with a Single Camera," *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, *2*:1403–1410 (Oct. 2003).

27. Diel D. D. *Stochastic constraints for vision-aided inertial navigation*. MS thesis, Massachussetts Institute of Technology, 2005.

28. Diel D. D., DeBitetto P., and Teller S. "Epipolar Constraints for Vision-Aided Inertial Navigation." *Proceedings of the IEEE Workshop on Motion and Video Computing (WAVC/MOTION '05)*. 2005.

29. Diesel J. and Dunn G. "GPS/IRS AIME: Certification for Sole Means and Solution to RF Interference." *Proceedings of the ION GPS-96, Satellite Division, Technical Meeting*. Kansas City, MO, 17-20 September 1996.

30. Diesel J. and King J. "Integration of Navigation Systems for Fault Detection, Exclusion, and Integrity Determination - Without WAAS." *Proceedings of the National Technical Meeting of The Institute of Navigation*. Anaheim, CA, 18-20 January 1995.

31. Diesel J. and Luu S. "GPS/IRS AIME: Calculation of Thresholds and Protection Radius Using Chi-Square Methods." *Proceedings of the ION GPS-95, Satellite Division, Technical Meeting*. Palm Springs, CA, 12-15 September 1995.

32. Do J.-Y., Rabinowitz M., and Enge P. "Multi-Fault Tolerant RAIM Algorithm for Hybrid GPS/TV Positioning." *Proceedings of the National Technical Meeting of The Institute of Navigation 2007*. San Diego, CA, 22-24 September 2007.

33. Durand J. M. and Caseau A. "GPS Availability Part II: Evaluation of State Probabilities for 21 Satellite and 24 Satellite Constellations." *Navigation, Vol. 37, No. 3*. Fall 1992.

34. Durrant-Whyte H. and Bailey T. "Simultaneous Localization and Mapping (SLAM): Part I," *IEEE Robotics & Automation Magazine*, *13*(2):99–108 (June 2006).

35. Ebcin S. and Veth M. "Tightly-Coupled Image-Aided Inertial Navigation Using the Unscented Kalman Filter." *Proceedings of the Institute of Navigation GNSS, 20th International Technical Meeting of The Satellite Division*. Fort Worth, TX, 18-20 September 2007.

36. Enge P. K., Vicksell F. B., and Goddard R. B. "Combining Pseudoranges From GPS and Loran-C for Air Navigation," *Proceedings of the IEEE*, 36–43 (1990).

37. Eustice R., Singh H., Leonard J., Walter M., and Ballard R. “Visually Navigating the RMS Titanic with SLAM Information Filters.” *Proceedings of Robotics: Science and Systems*. June 2005.

38. FAA , “Required Navigation Performance - RNP.” Federal Aviation Administration On-line Library, April 2007.

39. Farrell J. L. and Graas F. V. “Statistical Validation for GPS Integrity Test.” *Journal of the Institute of Navigation, Vol. 39, No. 2*. Summer 1992.

40. Fischler M. A. and Bolles R. C. “Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography,” *Communications of the ACM*, *24*(6):382–395 (June 1981).

41. Forsyth D. A. and Ponce J. *Computer Vision: A Modern Approach*. Upper Saddle River, NJ 07458: Prentice-Hall, Inc., 2003.

42. Giebner M. G. *Tightly-Coupled Image-Aided Inertial Navigation System via a Kalman Filter*. MS thesis, Graduate School of Engineering, Air Force Institute of Technology (AETC), Wright-Patterson AFB OH, March 2003. AFIT/GE/ENG/03-10.

43. Gordon I. and Lowe D. G. “Scene Modelling, Recognition, and racking with Invariant Image Features,” *International Symposium on Mixed and Augmented Reality*, 110–119 (2004).

44. Gratton L. and Pervan B. “Carrier Phase Airborne and Ground Monitors for Ionospheric Front Detection for Category III LAAS.” *Proceedings of the Institute of Navigation GNSS, 19th International Technical Meeting of The Satellite Division*. Fort Worth, TX, 26-29 September 2006.

45. Gray R. A. *An Integrated GPS/INS/BARO and Radar Altimeter System for Aircraft Precision Approach Landings*. MS thesis, AFIT/EN/ENG/94D-13, Graduate School of Engineering, Air Force Institute of Technology (AETC), Wright-Patterson AFB OH, December 1994.

46. Harris C. and Stephens M. "A Combined Corner and Edge Detector." *Alvey Conference*. 1988.

47. Heinrichs M., Hellwich O., and Rodehorst V. "Robust Spatio-Temporal Feature Tracking." *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVII. Part B3a,*. 2008.

48. Hofman-Wellenhof B., Legat K., and Wieser M. *Navigation: Principles of Positioning and Guidance*. Austria: Springer-Verlag Wien New York, 2003.

49. Holmes J. K. *Coherent Spread Spectrum Systems*. New York: Wiley-Interscience, 1982.

50. Hong , Sinpyoe. a. "Observability Analysis of GPS Aided INS." *Proceedings of the 2000 ION GPS-2000 Conference*. 2618–2624. Salt Lake City, UT: Institute of Navigation, September 2000.

51. Hoshizaki T., II D. A., Braun A., Mulyana A., and Bethel J. "Performance of Integrated Electro-Optical Navigation Systems." *Proceedings of the Institute of Navigation GPS/GNSS 2003*. 9-12 September 2003.

52. Huang G., Mourikis A. I., and Roumeliotis S. I. *Analysis and Improvement of the Consistency of Extended Kalman Filter Based SLAM*. Technical Report 2007-0001, Minneapolis, MN 55455: Department of Computer Science and Engineering, University of Minnesota, 2007. http://www.cs.umn.edu/~ghuang/paper/TR_consistency.pdf.

53. Hwang P. Y. and Brown R. G. "NIORAIM Integrity Monitoring Performance In Simultaneous Two-Fault Satellite Scenarios," *Proceedings of the Institute of Navigation GNSS, 18th International Technical Meeting of the Satellite Division*, 1760–1771 (Long, Beach, CA, 13-16 September 2005).

54. Hwang P. Y. and Brown R. G. "NIORAIM Integrity Monitoring Performance in Simultaneous Two-Fault Scenarios." *Proceedings of the Institute of Navigation GNSS*

18th International Technical Meeting of the Satellite Division. 13-16 September 2005.

55. Hwang P. Y. and Brown R. G. "From RAIM to NIOAIM: A New Integrity Approach to Integrated Multi-GNSS Systems," *InsideGNSS*, 24–33 (May-June 2008).

56. Jain A. K. *Fundamentals of Digital Image Processing*. Upper Saddle River, NJ: Prentice-Hall, 1989.

57. Jung S.-J., Eledath J., Johansson S., and Mathevon V. "Egomotion Estimation in Monocular Infra-Red Image Sequence for Night Vision Applications." *IEEE Workshop on Applications of Computer Vision (WACV '07)*. 2007.

58. Kalafus R. and Chin G. Y. "Performance Measures of Receiver Autonomous GPS Integrity Monitoring." *GPS Papers Published in Navigation (Red Book Series), Vol. V*. 1998.

59. Kaminer I., Pascoal A., and Kang W. "Integrated Vision/Inertial Navigation System Design Using Nonlinear Filtering," *Proceedings of the American Control Conference*, 1910–1914 (June 1999).

60. Kaplan E. *Understanding GPS: Principles and Application*. Norwood MA: Artech House, 1996.

61. Ke Y. and Sukthankar R. "PCA-SIFT: A More Distinctive Representation for Local Image Descriptors." *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2004.

62. Kelly R. J. "Linear Model, RNP, and the Near-Optimum Fault." *GPS Papers Published in Navigation (Red Book Series), Vol. V*. 1998.

63. Klette R., Schlüns K., and Koschan A. *Computer Vision: Three-Dimensional Data from Images*. Singapore: Springer-Verlag, Pte. Ltd, 1998.

64. Klippenstein J. and Zhang H. "Qualitative Evaluation of Feature Extractors for Bearing-Only Visual SLAM." *Proceedings of 2007 Canadian Conference on Computer and Robot Vision*. May 2007.

65. Larson C. D. and Raquet J. F. “The Impact of Attitude on Image-Based Integrity.” *Proceedings of the International Technical Meeting of the Institute of Navigation ION GNSS 2010*. San Diego, CA, 25-27 January 2010.

66. Larson C. D., Raquet J. F., and Veth M. J. “Developing a Framework for Image-Based Integrity.” *Proceedings of the 22nd International Technical Meeting of the Satellite Division of the Institute of Navigation ION GNSS 2009*. Savannah, GA, 22-25 September 2009.

67. Lee Y., VanDyke K., Decleene B., Studenny J., and Beckmann M. “Summary of RTCA SC-159 GPS Integrity Working Group Activities.” *Journal of the Institute of Navigation, Vol. 43, No. 3*. Fall 1996.

68. Lee Y. C. “Analysis of Range and Position Comparison Methods as a Means to Provide GPS Integrity to the User Receiver.” *Proceedings of The Institute of Navigation, 42nd Annual Meeting*. June 1986.

69. Lee Y. C. “Investigation of Extending Receiver Autonomous Integrity Monitoring (RAIM) to Combined Use of Galileo and Modernized GPS.” *Proceedings of the Institute of Navigation GNSS-2004, The 17th International Technical Meeting of the Satellite Division*. Long Beach, CA, 21-24 September 2004.

70. Lee Y. C. “Navigation System Integrity Using Redundant Measurements.” *Proceedings of the Institute of Navigation GNSS-2004, The 17th International Technical Meeting of the Satellite Division*. Long Beach, CA, 21-24 September 2004.

71. Lee Y. C. “A New Improved RAIM Method Based on the Optimally Weighted Average Solution Under the Assumption of a Single Fault.” *Proceedings of The Institute of Navigation National Technical Meeting*. Monterey, CA, 18-20 January 2006.

72. Lee Y. C. and O’Laughlin D. G., “A Performance Analysis of a Tightly Coupled GPS/Inertial System for Two Integrity Monitoring Methods.” Technical Paper, March 2000.

73. Linn G. W., Miller M. M., and Vallot L. C. "Embedded GPS/INS - The Advantage." *The 16th Biennial Test Symposium.* October 1993.

74. Longuet-Higgins H. "A Computer Algorithm for Reconstructing a Scene from Two Projections," *Nature*, (293):133–135 (September 1981).

75. Lowe D. G. "Object Recognition from Local Scale-Invariant Features," *IEEE International Conference on Computer Vision*, *2*:1150 (1999).

76. Lowe D. G. "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, *60(2)*:91–110 (2004).

77. Ma Y., Soatta S., Košecká J., and Sastry S. S. *An Invitation to 3-D Vision: From Images to Geometric Models*. New York, NY: Springer-Verlag New York, Inc., 2004.

78. Maofipoor S. "Updating the Navigation Parameters From the Image Sensor in a Multi-Sensor System." *Proceedings of the Institute of Navigation GNSS, 19th International Technical Meeting of The Satellite Division*. Fort Worth, TX, 26-29 September 2006.

79. Martini I. and Hein G. W. "An Integrity Monitoring Technique for Multiple Failures Detection," *Proceedings of the IEEE*, 450–467 (2006).

80. Maybeck P. S. *Stochastic Models, Estimation, and Control, I*. New York: Academic Press Inc., 1979. Republished, Arlington, VA: Navtech, 1994.

81. Maybeck P. S. *Stochastic Models, Estimation, and Control, II*. New York: Academic Press Inc., 1982. Republished, Arlington, VA: Navtech, 1994.

82. Miao G. J. and Clements M. A. *Digital Signal Processing and Statistical Classification*. Norwood, MA: Artech House, Inc., 2002.

83. Mikolajczyk K. and Schmid C. "A Performance Evaluation of Local Descriptors," *27*(10):1615–1630 (October 2005).

84. Misra P. and Enge P. *Global Positioning System: Signals, Measurements, and Performance*. Lincoln, MA: Ganga-Jamuna Press, 2001.

85. Moon , Sung Wooke. a. “Design and Implementation of an Efficient Tightly-Coupled GPS/INS Integration Scheme.” *Proceedings of the 2000 ION GPS-2000 Conference*. 159–165. Anaheim, CA: Institute of Navigation, January 2000.

86. Mourikis A., Trawny N., Roumeliotis S., Johnson A., and Matthies L. “Vision-Aided Inertial Navigation for Precise Planetary Landing: Analysis and Experiments.” *Proceedings of Robotics: Science and Systems*. Atlanta, GA, 27-30 June 2007.

87. Mourikis A. I. and Roumeliotis S. I. “A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation.” *IEEE International Conference on Robotics and Automation (ICRA)*. Roma, Italy, 10-14 April 2007.

88. Nielson M. B. *Development and Flight Test of a Robust Optical-Inertial Navigation System Using Low-Cost Sensors*. MS thesis, Graduate School of Engineering, Air Force Institute of Technology (AETC), Wright-Patterson AFB OH, March 2008. AFIT/GE/ENG/08-19.

89. Nikiforov I. and Roturier B. “Statistical Aalysis of Different RAIM Schemes.” *ION GPS-2002*. Portland, OR, 24-27 September 2002.

90. Nikiforov I. and Roturier B. “Advanced RAIM Algorithms: First Results.” *Proceedings of the Institute of Navigation GNSS, 18th International Technical Meeting of the Satellite Division*. Long Beach, CA, 13-16 September 2005.

91. Ober P. B. “RAIM Performance: How Algorithms Differ.” *Proceedings of the Institute of Navigation GNSS, 11th International Technical Meeting of the Satellite Division*. Nashville, TN, 15-18 September 1998.

92. Ober P. B. *Integrity Prediction and Monitoring of Navigation Systems*. Integricom Publishers, February 2003.

93. Ober P. B. and Harriman D. “On the Use of Multiconstellation RAIM for Aircraft Approaches.” *Proceedings of the Institute of Navigation GNSS, 19th International Technical Meeting of the Satellite Division*. Fort Worth, TX, 26-29 September 2006.

94. Ober P. B., Harriman D., and Wilde J. "Augur: RAIM for Dummies." *Proceedings of the IEEE, Aerospace Conference, Vol. 2*. 1999.

95. Parkinson B. W. and Axelrad P. "Autonomous GPS Integrity Monitoring Using the Pseudorange Residual." *Journal of the Institute of Navigation, Vol. 35, No. 2*. Summer 1988.

96. Potter J. E. and Suman M. C. "Thresholdless Redundancy Management with Arrays of Skewed Instruments," *Advisory Group for Aerospace Research & Development: Integrity in Electronic Flight Control Systems (AGARDograph 224)*, (AGARD-AG-224):15:1–25 (April 1977).

97. Pratt W. K. *Digital Image Processing*. Hoboken, NJ: John Wiley & Sons, Inc, 2007.

98. Pullen S. P., Pervan B. S., and Parkinson B. W. "A New Approach to GPS Integrity Monitoring Using Prior Probability Models and Optimal Threshold Search." *IEEE Position Location and Navigation Symposium*. April 1994.

99. Raquet J. F. and Giebner M. "Navigation Using Optical Measurements of Objects at Unknown Locations." *The Institute of Navigation 59th Annual Meeting and CIGTF 22nd Guidance Test Symposium*. Albuquerque, NM, 23-25 June 2003.

100. Rogers R. M. *Applied Mathematics in Integrated Navigation Systems, Second Edition*. Reston VA: American Institute of Aeronautics and Astronautics Inc., 2000.

101. Rosten E. and Drummond T. "Fusing points and lines for high performance tracking.." *IEEE International Conference on Computer Vision2*. 1508–1511. October 2005.

102. Sakai T., Fukushima S., Takeichi N., and Ito K. "Augmentation Performance of QZSS L1-SAIF Signal." *Proceedings of the Institute of Navigation National Technical Meeting*. San Diego, CA, 22-24 January 2007.

103. Schempp T., Peck S., Chou W., Lopez E., and Hendrich R. "WAAS Performance Improvements as a Result of WAAS Expansion." *Proceedings of the Institute of Nav-*

igation GNSS, 19th International Technical Meeting of The Satellite Division. Fort Worth, TX, 26-29 September 2006.

104. Se S., Lowe D., and Little J. "Vision-based mobile robot localization and mapping using scale-invariant features," *Proceedings of IEEE International Conference on Robotics and Automation (ICRA), 2001*, *2*:2051–2058 vol.2 (May 2001).

105. Shi J. and Tomasi C. "Good Features to Track." *IEEE Computer Vision and Pattern Recognition (CVR)*. 21-23 June 1994.

106. Soon B. K. and Scheding S. "Sensitivity Analysis of an Integrated Vision and Inertial System," *Position, Location, And Navigation Symposium, 2006 IEEE/ION, (IEEE Paper Number 0-7803-9454-2)*, 722–729 (25-27, 2006).

107. Spilker , James J.J. *Digital Communications by Satellite*. Englewood Cliffs NJ: Prentice Hall, 1977.

108. Strang G. *Linear Algebra and its Applications, Third Edition*. San Diego: Harcourt, Inc., 1988.

109. Strelow D. and Singh S. "Long-term Motion Estimation from Images." *International Symposium on Experimental Robotics (ISER 2006)*. July 2006.

110. Strurza M. A. "Navigation System Integrity Using Redundant Measurements." *Journal of the Institute of Navigation, Vol. 35, No. 1*. Winter 1988-1989.

111. Svedman M., Goncalves L., Karlsson N., Munich M., and Pirjanian P. "Structure from Stereo Vision using Unsynchronized Cameras for Simultaneous Localization and Mapping." *Proceedings of the International Conference on ntelligent Robots and Systems (IROS) 2005*. 2005.

112. Titterton D. H. and Weston J. L. *Strapdown Inertial Navigation Technology*. London: Peter Peregrinis Ltd., 1997.

113. Tomasi C. and Kanade T. *Detection and Tracking of Point Features*. Technical Report CMU-CS-91-132, Pittsburgh, PA: Carnegie Mellon University, April 1991.

114. Torr P. and Zisserman A. "MLESAC: A New RObust Estimator with Application to Estimating Image Geometry," *Computer Vision and Image Understanding*, *78*(1):138–156 (2000).

115. Trawny N., Mourikis A. I., Roumeliotis S. I., Johnson A. E., and Montgomery J. "Vision-Aided Inertial Navigation for Pin-Point Landing using Observations of Mapped Landmarks," *Journal of Field Robotics*, *24*(5):357–378 (April 2007).

116. van Diggelen F. and Brown A. "Mathematical Aspects of GPS RAIM." *IEEE Position Location and Navigation Symposium*. April 1994.

117. van Graas F. and Farrell J. L. "Baseline Fault Detection and Exclusion Algorithm." *Proceedings of the 49th Annual Meeting of The Institute of Navigation*. Cambridge, MA, 21-23 June 1993.

118. Veerman H. P. and Rosenthal P. "EGNOS Flight Trials, Evaluation of EGNOS Performance and Prospects." *Proceedings of the Institute of Navigation National Technical Meeting*. Monterey, CA, 25-28 September 2006.

119. Veth M. J. *Fusion of Imaging and Inertial Sensors for Navigation*. PhD dissertation, Graduate School of Engineering, Air Force Institute of Technology (AETC), Wright-Patterson AFB OH, April 2006. AFIT/DS/ENG/06-09.

120. Veth M. J. and Pachter M. "Correspondence Search Mitigation Using Feature Space Anti-Aliasing." *Proceedings of the Institute of Navigation GNSS, 63rd Annual Meeting*. April 2007.

121. Veth M. J. and Raquet J. F. "Fusion of Low-Cost Imaging and Inertial Sensors for Navigation." *Proceedings of the Institute of Navigation GNSS, 19th International Technical Meeting of the Satellite Division*. September 2006.

122. Veth M. J. and Raquet J. F. "Two-Dimensional Stochastic Processes for Tight Integration of Optical and Inertial Sensors." *Proceedings of The Institute of Navigation 2006 National Technical Meeting*. January 2006.

123. Veth M. J., Raquet J. F., and Pachter M. “Stochastic Constraints for Efficient Image Correspondence Search,” *IEEE Transactions on Aerospace and Electronic Systems*, *42*(3):973–982 (July 2006).

124. Walter T. and Enge P. “Weighted RAIM for Precision Approach.” *Proceedings of the Institute of Navigation GPS-95, 8th International Technical Meeting of The Satellite Division*. Palm Springs, CA, 12-15 September 1995.

125. White N. A. *MMAE Detection of Interference/Jamming and Spoofing in a DGPS-Aided INS*. MS thesis, AFIT/GE/ENG/96D-21, Graduate School of Engineering, Air Force Institute of Technology (AETC), Wright-Patterson AFB OH, December 1996.

126. Winkler S., Schulz H., Buschmann M., Kordes T., and Vorsman P. “Space-Time Adaptive Processing with Sparse Antenna Arrays.” *Proceedings of the Institute of Navigation GNSS, 17th International Technical Meeting of the Satellite Division*. Long Beach, CA, 21-24 September 2004.

127. Wu A. D., Johnson E. N., and Proctor A. A. “Vision Aided nertial Navigation for Flight Control.” *AIAA Guidance, Navigation, and Control Conference and Exhibit*. San Francisco, CA, 15-18 August 2005.

128. You S., Neumann U., and Azuma R. “Hybrid Inertial and Vision Tracking for Augmented Reality Registration.” *IEEE Virtual Reality Conference*. Houston, Texas,13-17 March 1999.

129. Young R. S. and McGraw G. A. “Fault Detection and Exclusion Using Normalized Solution Separation Methods.” *Proceedings of the Institute of Navigation GPS-2002, International Technical Meeting of The Satellite Division*. Portland, OR, 24-27 September 2002.

130. Young R. S., McGraw G. A., and Driscoll B. T. “Investigation and Comparison of Horizontal Protection Level and Horizontal Uncertainty Level in FDE Algorithms.” *Proceedings of the ION GPS-96, Satellite Division, Technical Meeting*. Kansas City, MO, 17-20 September 1996.

131. Zemskov E. and Nurmi J. “Yet Another FDE for Multiple Outliers.” *Proceedings of the National Technical Meeting of The Institute of Navigation 2007*. San Diego, CA, 22-24 September 2007.

132. Zhang Z. “A Flexible New Technique for Camera Calibration,” *IEEE Transactions on pattern Analysis and Machine Intelligence*, 1330–1334 (November 2000). Vol. 22, No. 11.

133. Zhang Z., Deriche R., Fauregas O., and Luong Q.-T. “A Robust Technique for Matching Two Uncalibrated Images Through the Recovery of the Unknown Epipolar Geometry,” *Artificial Intelligence (78)*, 87–119 (December 1995). INRIA Sophia-Antipolis, France.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)* 01–06–2010	**2. REPORT TYPE** Doctoral Dissertation	**3. DATES COVERED** *(From — To)* September 2006-June 2010

4. TITLE AND SUBTITLE

An Integrity Framework for Image-Based Navigation Systems

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Larson, Craig D., Captain, USAF

5d. PROJECT NUMBER

ENG 09-319

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765 DSN: 785-3636

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/DEE/ENG/10-03

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Air Force Research Laboratory
Attn: AFRL/RYRN (Dr. Jacob Campbell)
2241 Avionics Circle, Bldg 620
WPAFB OH 45433-7734
(937)255-6127 (DSN: 785-6127)
jacob.campbell@wpafb.af.mil

10. SPONSOR/MONITOR'S ACRONYM(S)

AFRL/RYRN

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

13. SUPPLEMENTARY NOTES

This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States

14. ABSTRACT

This work first examines fundamental differences between measurement models established for GPS and those of proposed image-based navigation systems. In contrast to single value per satellite GPS pseudorange measurements, image measurements are inherently angle-based and represent pixel coordinate pairs for each mapped target. Thus, in the image-based case, special consideration must be given to the units of the transformations between the states and measurements, and also to the fact that multiple rows of the observation matrix relate to particular error states. An algorithm is developed to instantiate a framework for image-based integrity analogous to that of GPS RAIM. The algorithm is applied cases where the navigation system is estimating position only and then extended to cases where both position and attitude estimation is required. Detailed analysis demonstrates the impact of angular error on a single pixel pair measurement and comparisons from both estimation scenario results show that, from an integrity perspective, there is significant benefit in having known attitude information. Additional work demonstrates the impact of pixel pair measurement relative geometries on system integrity, showing potential improvement in image-based integrity through screening and adding measurements, when available, to the navigation system solution.

15. SUBJECT TERMS

Integrity, Image-Based Navigation, Image-Aiding, Alternative Navigation Methods

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Dr. John F. Raquet (ENG)
U	U	U	UU	233	**19b. TELEPHONE NUMBER** *(include area code)* (937)255-3636x4580; email:john.raquet@afit.edu